UNIVERCITY OF LISBON
Faculty of Sciences
Department of Informatics

PRIVACY PRESERVING PROTOCOLS FOR SMART METERS AND ELECTRIC VEHICLES

André da Cunha Araújo de Jesus

Work supervised by Prof. Nuno Fuentecilla Maia Ferreira Neves

DISSERTATION

MASTERS ON SECURITY INFORMATICS

2015

Acknowledgments

The realization of this dissertation had important support and incentives without it would not have become a reality and to which I will be eternally grateful.

To Professor Nuno Ferreira Neves, for your guidance, full support, availability, opinions and criticisms, cooperation in resolving questions and problems that have emerged over the completion of this work and for all the incentive. Thank you.

To Vera who has always been by my side, for the help with revising, the companionship, strength, support and encouragement in the difficult times.

To Filipa, for the help with revising, and the friendship and companionship.

To Radu, Rodrigo, Bruno and others who do not have their names mentioned but surely they know, for the friendship and companionship.

Finally, knowing that alone none of this would have been possible, my special thanks to my parents, because they provide everything I needed, for their unconditional love, support and encouragement.

Funding

This work was partially supported by the EC through project FP7 SEGRID (607109), by national funds of Fundação para a Ciência e a Tecnologia (FCT) through project UID/CEC/00408/2013 (LaSIGE).

To all who accompany me in the journey of life.

Resumo

Actualmente existe a tendência para se adicionar mais inteligência em vários pontos da rede eléctrica, permitindo uma comunicação bidireccional entre a empresa fornecedora de energia eléctrica e as nossas casas. Ao longo dos próximos anos, os contadores de energia nas nossas casas serão gradualmente substituídos por um equipamento com mais capacidades, denominado medidor inteligente.

Os medidores inteligentes podem colher informações sobre os gastos de energia em tempo real, e encaminhar os dados para o fornecedor. Além disso, podem receber comandos do fornecedor (ou outros intervenientes) e agir em conformidade, nomeadamente através da interacção com equipamentos locais (por exemplo, ar condicionado ou congelador) para ajustar o seu modo de operação, diminuindo temporariamente o consumo de energia. Os medidores inteligentes podem ainda apoiar a produção local de energia (com painéis solares ou geradores eólicos) e o seu armazenamento (através de um banco de baterias ou veículo eléctrico), sendo necessário haver coordenação entre a sua operação e as empresas fornecedoras de energia eléctrica.

Estes medidores, quando coordenados de uma forma apropriada, podem permitir uma redução dos picos globais de consumo. Deste modo evitam investimentos na rede energética direccionados para lidar com estas condições extremas, que tendem a ocorrer durante o horário laboral. A evolução no uso de veículos eléctricos irá gerar também um grande consumo de energia. Caso todos os veículos se tornem eléctricos, a rede actual não tem capacidade para lidar com o enorme pico gerado. No entanto, estes veículos poderão também ter a capacidade de transferir para a rede parte da sua energia, o que significa que, poderão ser usados em caso de necessidade para colmatar flutuações no consumo de energia (juntamente com outras fontes alternativas de geração). Esta coordenação, quando eficiente, pode permitir grandes vantagens em situações limite, como por exemplo quando há um fornecimento reduzido de energia, em que os medidores podem desactivar total ou parcialmente os aparelhos domésticos, permitindo uma melhor distribuição de energia por todos, priorizando, se necessário, certos locais como por exemplo hospitais.

Como esperado, este tipo de configuração é propenso a muitas formas de ataque, desde a espionagem de comunicações até à manipulação física dos medidores inteligen-

tes. Por isso, é necessário desenvolver protocolos seguros que possam ser usados para proteger os dispositivos e aplicações que irão operar na rede eléctrica futura. Este projecto em particular, desenvolve uma solução que protege as comunicações entre o medidor inteligente e a empresa distribuidora de energia no que diz respeito aos ataques à privacidade. Nestes ataques, o adversário obtém informação sobre o que o utilizador está a fazer em sua casa, monitorizando em tempo real a informação que é transmitida pelo medidor inteligente.

Nos últimos anos tem-se assistido igualmente a uma evolução rápida nas tecnologias de transferência de energia sem fios, existindo actualmente alguns protótipos em funcionamento, como o carregamento de baterias em autocarros eléctricos numa universidade da Coreia do Sul. Uma eventual utilização generalizada desta tecnologia obriga à definição de novas formas de pagamento, possibilitando que os veículos eléctricos se possam abastecer em movimento. Se existir um protocolo demasiado simples que faça esta tarefa, pode levar a que o condutor seja identificado quando e onde carregar as baterias do seu veículo, algo que não acontece com um tradicional abastecimento de combustível pago com notas ou moedas.

Este projecto lida com duas vertentes relacionadas que tratam da aferição do consumo de energia. Uma é baseada nos contadores inteligentes das casas, e outra nos "contadores" em veículos (mais concretamente, a forma de pagamento da energia transferida sem fios para um veículo em movimento). Apresentam-se diferentes técnicas/algoritmos já propostos que podem contribuir para uma solução, mas que apesar disso não conseguem atingir todos os requisitos e funcionalidades pretendidas de forma isolada. Estabelece-se também uma relação com o trabalho já realizado que utiliza tais técnicas.

É estudado um protocolo especifico, o Low Overhead PrivAcy (LOPA), que organiza vários medidores num grupo. Em cada grupo é gerada secretamente uma chave entre cada medidor do grupo, depois é criada a partir dessa chave uma outra chave, que é somada a cada medição que cada medidor envia para um agregador, sem que ninguém consiga ver o valor da medição individual (devido à chave). O agregador, ao somar todas as medições de todos os medidores de um grupo, obtém o valor total de consumo de todos os medidores. O agregador, no entanto, não consegue saber cada medição individual, devido ao modo como a chave é gerada, garantindo a privacidade de cada casa. Este protocolo é explicado em detalhe, implementado e avaliado.

São propostos também três protocolos para o pagamento da transferência de energia, que permitem manter o anonimato de um veículo, evitando que se saiba quando ou onde este circula. Os protocolos também lidam com ineficiências de transmissão, assegurando uma rapidez, simplicidade e segurança adequadas para serem aplicados em carros em movimento a velocidades habituais de circulação. Um dos protocolos permite uma transferência de energia pós-paga, e os outros dois usam uma modalidade de pré-pagamento,

um com contas temporárias e o outro com dinheiro digital. Estes protocolos baseiam-se num conjunto de mensagens que empregam técnicas como assinaturas digitais (para garantir a integridade e autenticação das comunicações), técnicas de cifra, dinheiro digital, ou entidades terceiras confiáveis para permitir a confidencialidade. Pretende-se que seja assegurada a segurança do pagamento, ao mesmo tempo que é permitido ao ponto de carregamento identificar o responsável pelo veículo, em caso de incumprimento. O dinheiro digital e o protocolo de perfis pseudo-anónimos foram implementados e avaliados em duas plataformas diferentes. Os resultados experimentais foram muito satisfatórios, dando indicações de que estes protocolos poderiam ser utilizados na prática.

Palavras-chave: Privacidade, Rede inteligente, Medidores inteligentes, Carregamento de veículos em movimento

Abstract

There is currently a trend to add more intelligence to various points of the electric grid, thus enabling a bidirectional communication path between the electrical utility company and our homes, by upgrading the existing components along the way. For example, the metering devices in our homes will be gradually replaced with a more capable equipment, called smart meter. Smart meters can collect information about energy spending in real-time, and forward this data to the utility. Moreover, they can receive information from the utility (or other operators) and act on it, for instance, by interacting with local equipments (e.g., air conditioner or refrigerator) to adjust their operation mode (e.g., make them decrease the energy use). Smart meters can also support local energy production (e.g., solar panels or windmills) and storage (e.g., batteries), by coordinating its operation with the utility companies.

As expected, this sort of setting is prone to many forms of attack, ranging from eavesdropping on the communications to the physical tampering of the smart meters. Therefore, it is necessary to develop secure protocols that can be used to protect the devices and applications that will be operating in this future smart grid. In particular, in this project we study and evaluate a solution that protects the communications between the smart meter and the electrical company with respect to attacks on privacy. For instance, it addresses a form of attack where the adversary learns information about what a person is doing at home by monitoring the messages transmitted by the smart meter in real-time.

In recent years there have been rapid developments in Wireless Power Transfer technology (WPT). There are currently some prototypes in operation, such as charging batteries in electric buses at a university in South Korea. In the event of a widespread use of this technology, it is required that new forms of accounting and payment of energy are established. This project proposes a protocol for the payment of energy transfer that ensures the anonymity of the vehicle, precluding attacks that attempt to determine where it circulates. The protocol also handles transmission inefficiencies, ensuring a fast, simple and adequate application in cars moving at normal speeds of movement.

Keywords: Smart Grid, Smart Meters, Privacy, Charging vehicles on the move

Contents

List of Figures **xiii**

List of Tables **xv**

1 Introduction **1**

1.1 Smart meters data transmission 3

1.2 Electrical vehicle energy purchase 5

1.3 Contributions 6

1.4 Work scheduling 7

1.5 Structure 9

2 Context and related work **11**

2.1 Related work on privacy on smart meters 13

2.1.1 Zero Proof Knowledge 13

2.1.2 Differential privacy 14

2.1.3 Anonymization 15

2.1.4 Homomorphic encryption 16

2.1.5 Trusted computing base and trusted third party 16

2.1.6 Battery based solutions 17

2.2 Related work on WPT payment 18

3 Low overhead protocol **21**

3.1 The LOPA protocol 21

3.2 Implementation and evaluation 23

3.3 Summary of the findings 27

4 Electrical vehicle energy purchase **29**

4.1 Payment system structure 29

4.1.1 System model 29

4.1.2 Basis and assumptions 31

4.1.3 Price Broadcaster 32

4.2 Payment Protocols 33

4.2.1 PP – Pseudoanonymous Profiles 34

4.2.2 APP – Anonymous Pre-Payment 36

4.2.3 Generation and security of digital notes 38

4.2.4 ADD – Anonymous Digital Money 41

4.3 Implementation 43

4.3.1 Evaluation 45

5 Conclusions **49**

References **56**

List of Figures

1.1 Hourly consumption 2
1.2 Smart meters examples 3
1.3 Photo of protest against smart meters 5
1.4 Charging points 6
1.5 First Gantt diagram 8
1.6 Final Gantt diagram 8

2.1 Zero Proof Knowledge 13
2.2 Example of normal distribution 14
2.3 Example of anonymization based on a trusted third party (TTP). 15
2.4 Example of a battery based solution for privacy. 17

3.1 Example of a key for each meter pair 22
3.2 The general idea behind DH 23
3.3 The structure of an hash function 24
3.4 The organization of the LOPA implementation. 25

4.1 System entities 30
4.2 General structure of the AES algorithm 43
4.3 The main calculations of a blind RSA signature 45

List of Tables

2.1 Time requirement for a transaction 19

3.1 Example of the list of $K_{m,k}$ 22

3.2 Simplified example of X_m^i key generation and encryption. 23

3.3 LOPA run time 26

4.1 Message of charging pricing 33

4.2 Pseudoanonymous profiles protocol example 34

4.3 Anonymous pre-payment protocol example 36

4.4 TCB proving message 37

4.5 Identification list example 39

4.6 Composition of a digital money bill 40

4.7 Anonymous digital money protocol example 42

4.8 The main calculations of the RSA algorithm 44

4.9 Time in milliseconds for different steps related to digital notes 46

4.10 Time in milliseconds for each step of the PP protocol 47

Chapter 1

Introduction

Society as it is organized nowadays has the need for several basic resources such as gas, water and electricity. These basic resources are usually distributed to all buildings, whether residential or industrial. However, these resources are not easy to capture, either by scarcity or the high costs associated. For example, in energy, there are more and more environmental regulations that restrict certain forms of gasoline production. Another example is the cost of building a power plant, which can be very high. Besides, it may need many permits to ensure, for instance, that a dam does not flood a village, or that a nuclear power plant properly takes care of its waste. In addition, there can be discontent of the local population, which can harden the creation of power plants.

Another management problem is related to the high variation in consumption. For example, in the summer, when the hydroelectric plants produce less power, there is an increased use of air conditioners. Domestic consumption peaks usually match the time interval before and after the normal working hours. Another issue will arise with electric vehicles (EV), which may also cause a peak of power consumption when charging. On the other hand, they may also have the ability to sell back energy to the network, which can be helpful in periods of highest energy usage.

The micro and mini power production is a growing trend, especially because the acquisition costs are lowering and there are government incentives, but they have the downside of having an irregular production. For example, a photovoltaic panel is greatly affected by the climate (works ideally under clear sun). The traditional electricity production also faces challenges, like the consumption of fossil fuels being restricted by environmental regulations. Moreover, it especially affects countries that do not extract oil, leading to a dependence on others, which often causes difficulties during fuel crises.

Consumers have increasing necessity of resources. Even if a region is partly dis-

connected from the rest of the network, consumers still need energy, with near 100% availability. Energy shortages lead to economic loss, for example stopped businesses, spoiled food, etc.

There are other extreme situations that happen once in a while, such as the blackout that occurred in the northern United States and Canada (in August 2003) [1], the energy crisis in Japan (e.g., Fukushima accident [2]) among numerous other smaller blackouts. This needs to be addressed, and there should be a way to prioritize the delivery of resources in this kind of situations (e.g., to hospitals).

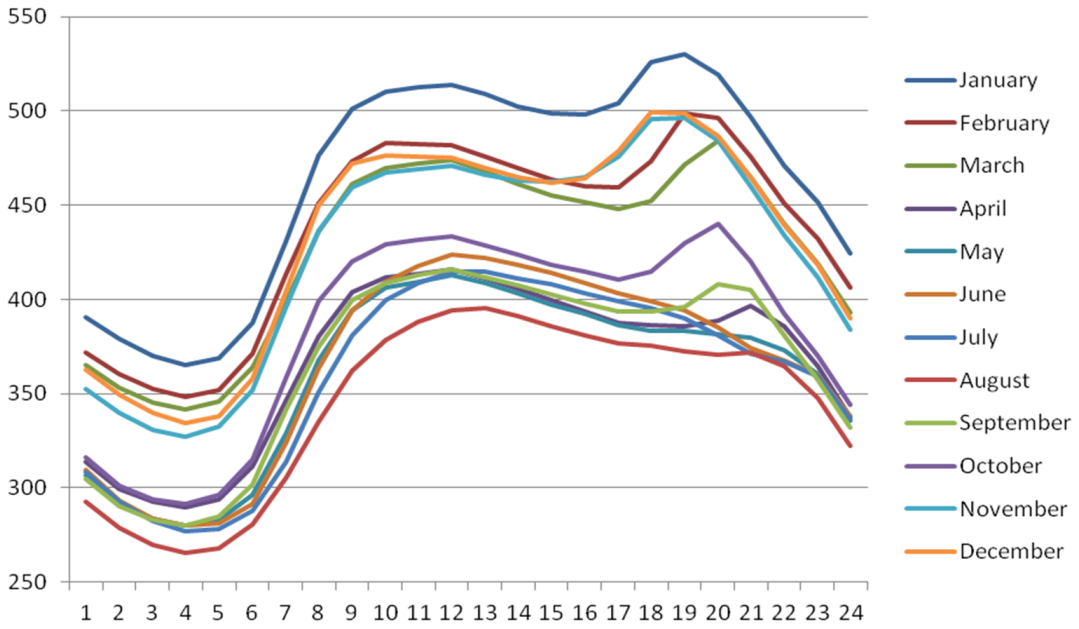

Figure 1.1: Hourly consumption (in MW) at each 3rd Wednesday of every month from all EU countries in the year 2013 [3].

A smart grid can efficiently distribute an insufficient energy source to where it is needed, taking advantage of small producers and implementing other measures, such as commanding vehicle charging points to start buying energy from vehicles. Nowadays, the consumption peaks are typically addressed by spare generation plants that are connected to the grid during these peak loads, which is normally inefficient and expensive. One of the main jobs of the smart grid is to reduce these consumption peaks, which besides increasing availability, leads to decreased costs. In Figure 1.1 it is shown the aggregated hourly consumption of all EU countries in 2013 at each 3rd Wednesday of every month. This shows that energy demand is far from static, not only changing during the day but also in every season.

In case there is a failure in the structure (e.g., an area becomes disconnected due to a natural disaster), the smart grid can recover from this situation more effectively by

resorting to local energy production (e.g., solar panels) and stored energy in the electrical vehicles.

1.1 Smart meters data transmission

The normal meters are being replaced with meters that are more capable and can bidirectionally communicate with the energy provider or other entities. A smart meter can be seen on Figure 1.2. Aesthetically they do not differ much from the current (non-smart) meters.

Smart meters will allow a reduction of manual readings by a person, by automatically transmitting the consumption data to the electrical distribution operator. This supports more frequent readings, eliminating the use of estimates, which can lead to unexpected price adjustments for some consumers. Besides, if consumers know their consumption accurately, it might encourage savings, especially if costs change during the day. For example, the user might turn on the washing machine at night because power is cheaper at that time.

Figure 1.2: Two examples of smart meters, left from [4] and right from [5].

One approach to identify theft or losses in the power distribution is by installing a meter at certain points of the network (e.g., secondary electrical substations). The values

collected by these meters are then compared to the sum of the readings of all smart meters connected to that point, and if there is a difference in consumption, it could indicate theft or loss.

If smart meters send measurements at a very fast pace (e.g., every few seconds or less), these precise measurements may allow the inference of what is being done inside that building or home. For example, a lamp has a linear consumption, but on the other hand, a fridge consumes energy for a short time and then disconnects for a longer period. With this information, it is possible to know if someone is at home, and even to characterize people through their consumption habits, which can raise privacy concerns. More complex information can be acquired through consumption analysis. For instance, it is possible to know to a certain degree of sureness, in some televisions, what program is being watched (adult programs, copyrighted material, political or religious shows, etc) [6].

When electrical vehicles start buying energy from the grid, there is the possibility to identify the vehicle brand by the amount of energy bought (as this correlates to the amount of energy it can carry). In addition, with an EV, it might be possible to know where the vehicle is located when it is charging/selling, which indirectly could allow the tracking of a certain person.

The data from meters is read by the utility company and other entities/components. Examples are data aggregators, power substations or other third parties involved at some point, and all these entities/components could violate privacy. It is actually much easier to compromise people's privacy by watching their resources consumption than by other means of surveillance. An attacker might also listen to the data being send, and after an inference process, decide on the best course of action to perform some malicious activity.

In some implementations of the smart grid, there is already resistance to its adoption. People are being alerted to the risk of privacy loss, thus creating a real need for specific legislation on the area. On Figure 1.3 is displayed a photo of a protest against the installation of smart meters.

As it is usually the utility company or distributor to install the smart meter, privacy is a second thought feature, however fraud detection is a priority. These two features may look incompatible at first sight, but their combination is possible as will be shown further ahead.

Figure 1.3: Photo of protest against smart meters (from [7]).

1.2 Electrical vehicle energy purchase

Another issue concerning energy is that, over the past few years, there have been attempts to find alternatives for people mobility, and solutions that avoid the traditional internal combustion engines in vehicles. There are already in the market vehicles that have an electric motor and batteries.

Purely electric vehicles need to charge periodically their batteries, usually from an electrical outlet in the garage or from power charging points around the city. Currently, batteries still do not have the energy density of traditional fuel and they are relatively expensive. In addition, batteries undergo a natural degradation (with and without use), which cause vehicles with only batteries to have a shorter autonomy than a typical internal combustion engine. The long charging time of current batteries is also a large psychological barrier in personal transportation and a real logistics barrier in commercial transport.

An alternative way to deal with this problem is to build vehicles with the ability to charge the batteries while they are moving. Wireless Power Transfer (WPT) is a technology currently under development, with some prototypes already available in KAIST University, South Korea. These prototypes look promising, reaching an efficiency of around 80% [8]. In Figure 1.4 it is shown a WPT charging point under construction and one already in operation.

With this charging method, the payment for the transferred energy must also be carried out through wireless communication, ideally without human intervention during the transfer process. However, any naive protocol will suffer from attempts at its integrity,

Figure 1.4: Charging points for WPT. On the left there is a charging point in construction and on the right one already working (from [9] and [10]).

energy stealing or compromise drivers privacy. These disadvantages are evident when compared to classic combustion engine vehicles, in which one can purchase fuel without revealing identity, the route or the traveled distances, because there is a cash payment method (notes/coins). Therefore, the energy purchase protocols must guarantee by default the privacy of the users. Another characteristic of these protocols is the need to carry out the transaction in a short period of time because the vehicle is moving, and there is a small time window to communicate, due to the wireless communication range limit.

1.3 Contributions

This work contributes firstly with an implementation and evaluation of the LOPA protocol, where smart meters are organized in groups. They generate a key between them, in a way that no eavesdropper nor other group of meters can know the key that each meter generated. Based on that key, new temporary keys are created to encrypt the meter readings. When all the encrypted readings are added at an aggregator, the total value will be the same as the sum of the actual readings. The protocol was evaluated on two different testbeds, one with a powerful Pc and another on a weak device (an old Raspberry Pi).

Then, the other main contribution of this work is the proposal of three protocols for the payment of WPT for EV. The structure, entities and requirements are laid. Then, the three different protocols are designed and explained. These protocols allow vehicle users to purchase energy, without revealing their real identification to the entity who sells energy. In addition, they assure that all energy purchased can be billed.

In the first protocol, the payment is made a certain time after the energy transfer, similar to the domestic energy purchase (could be a monthly payment). The other two

protocols, on the other hand, are pre-paid. One uses digital money, thus enabling the energy purchase, even if the charging point has no network connectivity (but has energy). The other uses account balances, similar to the rechargeable tickets for public transportation. Two of the protocols were evaluated on two different testbeds, one based on a Pc and another on a Raspberry Pi.

1.4 Work scheduling

This dissertation started in October 2014 and ended at July 2015. Included with the preliminary report, there was a Gantt chart defining the main tasks and their distribution through time. Figure 1.5 has a representation of this Gantt chart with the following tasks:

1. Definition of requirements and scenarios of the smart grid
2. Analysis of privacy enabling technologies on smart metering readings of the power consumption
3. Design of an algorithm that meets the requirements, namely about providing accurate readings and keeping the privacy of the users
4. Implementation of the algorithm and test scenarios
5. Testing the solution
6. Document the solution and results in a report

This work started focused on smart meters and the related privacy issues to the users. In particular, the short interval reading of smart meters was the original focus of work. Therefore, the LOPA protocol was selected to be implemented. The protocol was supposed to be tested on smart meters, but we had no access to them. Smart meters are not sold from a shelf, but they are ordered by electrical companies according to a given specification. Therefore, since we could not complete the implementation, we decided to focus on protocols for payment of WPT for EV. This lead to the creation of new tasks, which are represented in a new Gantt diagram in Figure 1.6. The new task list is as follows:

1. Definition of requirements and scenarios of the smart grid
2. Analysis of privacy enabling technologies on smart metering
3. Design of an algorithm that meets the requirements

4. Implementation of LOPA algorithm and testbeds setup
5. Evaluating the solution
6. Analysis of WPT payment protocols
7. Design of the protocols that meets the requirements (WPT payment protocols)
8. Implementation of payment protocol and testbeds setup
9. Evaluating the solution
10. Creation of an article for submission
11. Document the solution and results in a report

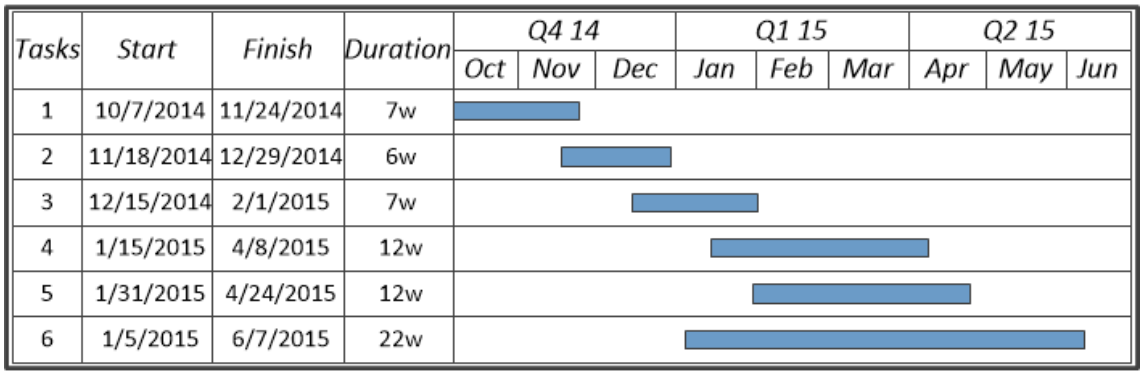

Figure 1.5: First Gantt diagram, delivered with the preliminary report.

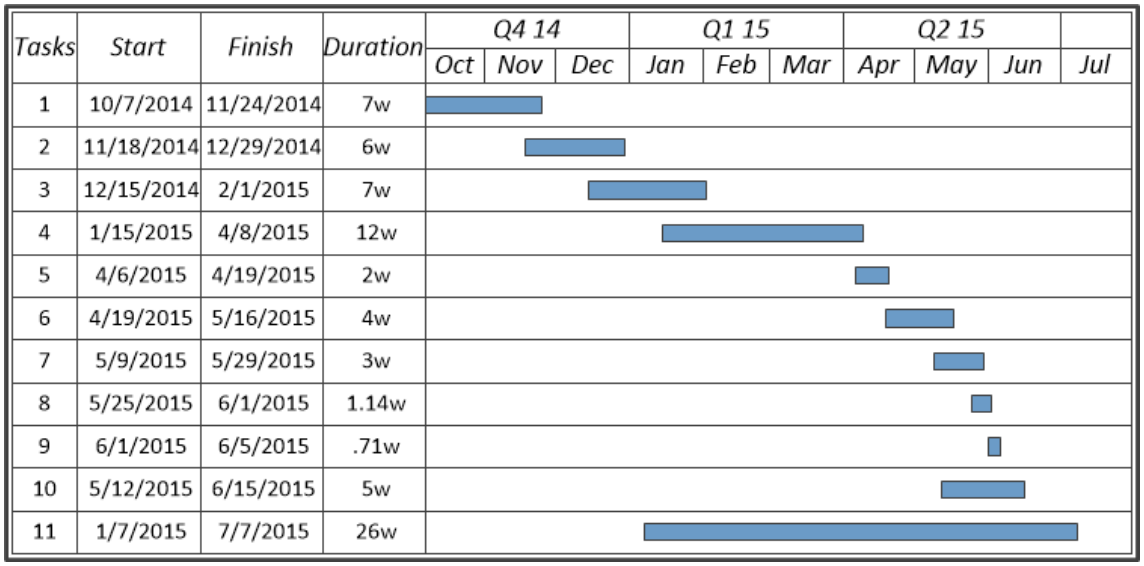

Figure 1.6: Final Gantt diagram, representing the actual distribution of tasks.

1.5 Structure

In the following chapter there are presented some requirements and functionalities that are expected from the smart grid. Then it is presented related work, which includes technological approaches as ingredients, that isolated are not enough to meet the privacy and requirements for the smart grid. In the following chapter, the Low Overhead Protocol is explained, implemented and evaluated. Then, in the next chapter, it is presented the payment system structure for the payments of WPT to EV. In addition, there are explained three energy payment protocols with very different characteristics, and details of each protocol are displayed, featuring the runtime values in two types of platforms. Thus, it is demonstrated that these allow the completion of the transaction at normal driving speeds. And finally the conclusion is presented.

The list of chapters is as follows:

- Context and related work
- Low overhead protocol
- Electrical vehicle energy purchase
- Conclusion

Chapter 2

Context and related work

The Smart Grid is a development of the existing power distribution infrastructure, which maintains existing features and introduces new ones. Among the features to be introduced are special energy plans, for example, reduced energy prices for certain industries or social tariffs. Moreover, there can be areas with priority for the available energy, such as hospitals or other places with bedridden patients on life support (e.g., medical ventilator). To support this sort of features the smart grid should be able to: send price lists to users; receive the monthly/weekly count (low frequency data); get the current consumption (may be obtained approximately in real time); and receive a promise of future consumption (the commitment involves a reward for accurate values). The grid should also provide information on the quality of service, including the electricity parameters (e.g., voltage). Future smart meters may measure more than just electricity consumption, as they could also collect values about other resources such as water, gas, and heat.

In case of selling energy to the grid, the price can vary depending on the hour or day of the week, but there can be some imbalance. In case many consumers are selling a great amount of energy in a more favorable time, it might lead to a big energy injection into the network that is not actually needed, and therefore not being efficient nor cost-effective. To address this imbalance, energy buying can also be done in an energy auction, therefore becoming easier to anticipate the energy consumption. There should be some kind of regulatory authority, which should look after (and certify) the various stakeholders, such as utility companies, price lists and others.

To accommodate for the above scenarios, certain equipment will have to be installed in the Smart Grid. For instance, if an electric vehicle goes to another house and sells energy at that point, the corresponding value of the energy should go to the owner of that car. The vehicle should be the one who manages its own batteries. If the vehicle owner is also the owner of the batteries (they are not being rented), he or she has interest in the

durability of the batteries. Typically, durability is reduced with charging and discharging cycles, meaning that this aspect should also be taken into account when making the decision about selling or not energy.

The network must be able to operate in case of a segmentation, namely in island mode. The island mode can be forced, for example, by turning off several lines after a disaster. In case of a disaster, the network should inform everyone of the event, and it should perform safety actions, like switching off gas appliances automatically in case of earthquake. In addition, in case of shortage of resources, priority of power delivery should be established, giving it first to hospitals or other critical services.

There should be the ability to encourage and/or impose a reduction of consumption at certain times, thus avoiding consumption peaks. The incentive and the imposition should be different processes. The first can be such as increasing the energy cost, and the second can be a reduction on the energy flow limit as a mean of enforcement.

Newer buildings tend to have more intelligence, including home automation and controls in various aspects, such as air conditioning, lighting, security, among others. The integration of these components is natural and allows energy savings, by decreasing the amount of different controllers, and installation cost reductions, by using fewer components. A Home Area Network (HAN) controller is a component that interacts with the equipment inside the house and the smart meter, which can manage the power consumption accordingly to the suggestion (or imposition) of the smart meter. Moreover, the HAN allows energy selling to the grid (e.g., based on installed batteries or some renewable source), taking into account the benefit of such transaction. This device may also allow the sharing of the consumer information in a controlled manner. For instance, in case of a reward being offered to compensate for the user data, the HAN controller allows features like directed advertising, pay per use insurance, and sending consumption statistics, at the cost of some loss of privacy. However, since the HAN belongs to the user, he or she can give directions on what information may be transmitted. This device may also control a set of batteries that allow energy obfuscating the actual consumption if privacy is not guaranteed to the user. It can also show what are the costs or profits that are being obtained with the usage of batteries. This controller should have its source code open, so that users can inspect and verify the code for its security. The HAN should be easy to use, especially for the people who do not usually deal with technology. In addition, it should be configurable to suit many different users needs.

The smart meters, HAN controllers, aggregators and other equipment should be quick to report and consume little power. However, they have a communication link that has high delay and low bandwidth available, and they have a small processing capacity.

2.1 Related work on privacy on smart meters

There are various approaches for consumers to have privacy when connected to the smart grid. And it is important an efficient and effective application of them. There are also various technologies and architectures trying to provide privacy, but they need to be correctly and fairly implemented by the right entities. There is the need of attention to the privacy matter, not only by the ones who lose it, but also by those who should ensure it.

2.1.1 Zero Proof Knowledge

Round	Alice		Bob
Keygen	$g, x \in Z^*_p$	$pk = g^x \bmod p, g, p$ →	pk, g, p
1	Pick $r \in Z^*_p$	$c = g^r \bmod p$ →	c
2		Please send r or $(x + r) \bmod (p - 1)$? ←	Flip a coin
3-a	If head	r_1 →	Check if $g^{r_1} \bmod p = ^? c$
3-b	otherwise	$z = (x + r) \bmod (p - 1)$ →	Check $g^z \bmod p = ^? c * pk$

Figure 2.1: Example of a Zero Proof Knowledge usage (inspired from [11])

Zero Proof Knowledge is a way for one entity (prover) to prove to another entity (verifier) that a given statement is true, without conveying any information apart from the fact that the statement is indeed true. This allows privacy and enforces honest behavior. One example of a Zero Proof Knowledge usage can be seen in Figure 2.1. There is a first commitment of Alice to the value x, which is never shown to Bob. She also sends a random value based on a r. Then, Bob asks for r or $(x + r)\ mod\ (p - 1)$, and then Alice answers accordingly.

Other example of usage is when smart meters want to send an aggregation of readings for billing purposes, but without revealing individual readings. The meter is the prover, the aggregator the verifier. The meter commits to the individual readings, then sums the individual readings and sends that value and the commitment, which can prove that the calculation is indeed true.

This kind of technique requires various rounds of communication if it is not non-interactive. Therefore, it can be relatively inefficient.

Non-interactive zero proof knowledge avoids various rounds of communication (in practice, with an hash function or a common reference string), but attention has to be given to the increased probability of attack, because in a hash function there could be a collision.

This kind of protocol might give proof of delivery, and offer verifiable computation, and it might help meters detect malicious aggregators.

This kind of approach is used in these works [12] [13] [14] [15] [16].

2.1.2 Differential privacy

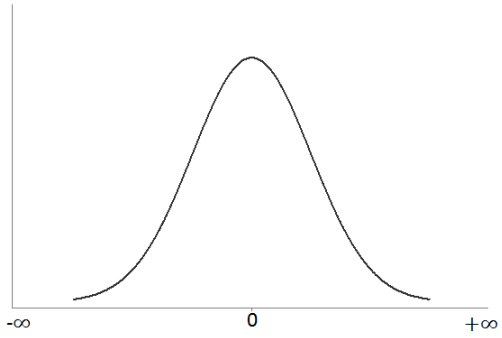

Figure 2.2: Example of normal distribution with $\mu = 0$ and $\sigma = 10000$.

Differential privacy consists in adding noise to the consumption data before sending it, and when all the data (with noise) from all meters is aggregated, the total value is near the actual sum.

For instance, in a possible implementation of differential privacy, each meter generates a random number, but that random number follows a normal distribution probability, for example with $\mu = 0$ and high value for σ (see Figure 2.2). Then, each meter adds this random number, which is the noise, to their readings. When all meter readings with noise are added, the total value will probabilistically be near the actual sum of meter readings without the noise.

With the noise added, it becomes more difficult or impossible to discover a fraudulent consumption or a distribution loss. To discover these problems, the aggregator must have a precise aggregated value. Moreover, it does not allow to determine if the meters are behaving correctly or not. This technique, however, claims perfect privacy over other approaches.

This kind of approach is used in these works [17] [18] [19] [20] [21] [22].

2.1.3 Anonymization

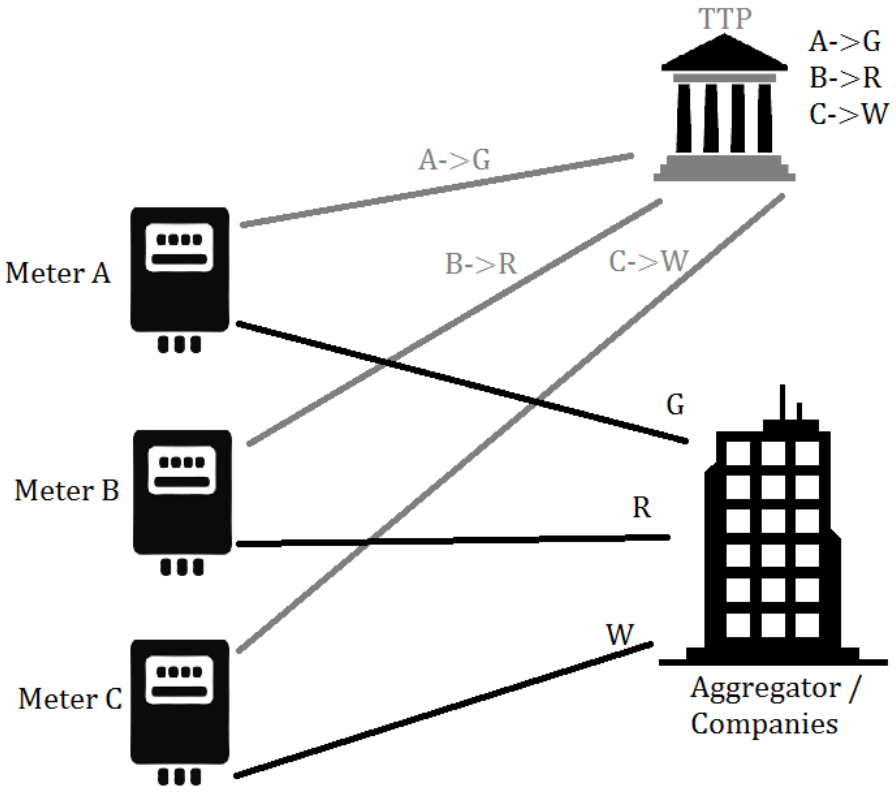

Figure 2.3: Example of anonymization based on a trusted third party (TTP).

Anonymization is in essence the difficulty or impossibility of associating a meter reading to a person/house/meter. In practice, the privacy property that is achieved is unlinkability between the meter reading and who sent it. This can be attained with the help of Third Trusted Parties (TTP) or Trusted Computing Bases (TCB). Nevertheless, with big data techniques, among others, anonymization techniques can be rendered ineffective, since sometimes it is possible to re-identify the owner of the data. This approach only works if the data does not need to be linked. For example, billing related data cannot be anonymized.

In Figure 2.3 it is shown an example of anonymization, based on a TTP. The aggregator only knows the identifications G, R and W instead of their true identities, A, B or C. Only the TTP knows the true association of identifications.

Another way to achieve anonymization would be simply to not provide an identification, but this could lead to abuse, thus this approach is not used alone.

This kind of approach is employed in these works [16] [22] [23] [24] [25].

2.1.4 Homomorphic encryption

This type of privacy preserving technique relies on the homomorphic property of encryptions or secret sharing schemes. A homomorphic encryption is an encryption scheme that allows addition (or other) operations to be performed over encrypted data, and when the data resulting from that operation is decrypted, it still keeps the addiction operation correctly applied. For example, the encryption algorithm E with key key would ensure that the following is true for all values: $\mathtt{E}_{key}\mathtt{[5]} + \mathtt{E}_{key}\mathtt{[2]} = \mathtt{E}_{key}\mathtt{[7]}$. The entity that does this operation does not need to decrypt individual values.

Individual reading values are encrypted among a group of meters and then joined (without decrypting). Then, the encrypted joined value is provided to the aggregator, and when it is decrypted the accumulated value becomes known.

This approach usually requires a much higher computational power, therefore causing a larger energy consumption, and requiring a bigger bandwidth due to increased cipher text blocks.

This kind of approach is used in these works [26] [27] [18] [28] [29] [30] [31] [32] [19] [20] [21].

2.1.5 Trusted computing base and trusted third party

Nowadays, traditional meters are the trusted devices to measure the consumption. They are secured with a hard to open package and seals that detect fraud, among other measures.

A TCB is a small hardware (or software) that is created with a special attention to security. It is often tamperproof, for example by placing the system on a physical chip. Its design and functionality should be really simple, in order to facilitate the demonstration of its security (less complexity leads to less errors). The TCB usually implements functions like encryption and decryption operations, digital signatures, hashing, key and certificate storage, storage of information, or other crucial operations. A TCB should do minimal tasks, and ideally, when a TCB is tampered, it is noticeable and actions can be taken.

A TCB is an important concept but, by design, it is a single point of failure, putting at risk the whole system. Replication with diverse designs might aid to avoid single points of failure.

One example of the usage of a TCB aiding privacy in smart metering, is the placement of a TCB on all smart meters. They digitally sign metering readings without providing any other kind of information (only the reading). This anonymous meter reading is sent to the aggregator, to verify the digital signature. The aggregator knows that it came from a TCB, but not which one, therefore privacy is assured.

A TTP has characteristics similar to a TCB, but in this case, instead of being a device it is a different entity doing part of the computation. All the other involved entities must trust this entity. Therefore, it must be independent of other entities. The TTP ends up being a single point of failure, and a rather strong assumption if too much trust is given.

On smart meters, a TTP can be used to allow unlinkability between readings and the meter, and maintain fraud/loss detection. This unlinkability can only be broken by the TTP itself, when for some reason it is required to provide information related to privacy.

In some implementations, the TTP might not be a truly third party as different countries have different rules. If the security of the TTP is broken, then it allows for mass surveillance.

This kind of approach is used in these works [26] [24] [25] [33] [27] [34].

2.1.6 Battery based solutions

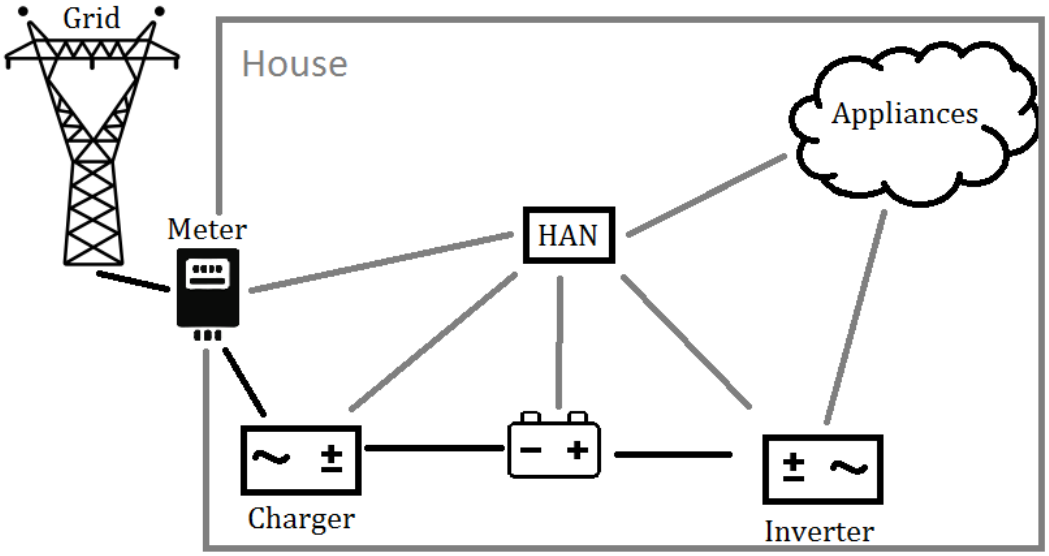

Figure 2.4: Example of a battery based solution for privacy.

A battery-based privacy solution is in essence a battery that soothes the consumption variance, thus providing privacy by eliminating the small variance of energy consumption

that is created by distinct devices (e.g., a fridge and lamp). Therefore, it becomes harder or impossible to distinguish different load signatures by observing the metering values.

This approach requires an upfront investment from the consumers. Therefore, privacy turns up as a paid feature and not a default characteristic. Nevertheless, this can become a viable approach if the smart grid and regulators do not care for the privacy of the consumers.

One way to recover part of the investment on batteries is for the consumer to generate energy or buy energy when it is cheaper. Nevertheless, batteries may require considerable space, and sometimes they have special requirements. For example, lead batteries require ventilated, waterproof spaces and a specially certified installation.

Batteries and energy conversion equipment have some degree of inefficiency, having a power loss associated with charging and discharging, or just storage throughout time, which constitutes another cost.

.

If a user already has an EV, it can be used to achieve the same solution, without the need to buy more batteries and the corresponding electronics. Of course, in this case, the privacy protection could only be available when the EV is connected to the house.

This kind of approach is used in these works [35] [36] [37] [38] [39].

2.2 Related work on WPT payment

There are several protocols in the literature that allow authentication among the entities involved in the transaction, and various payment structures. However, in the case of charging moving vehicles, there are requirements related to privacy, the speed of the EV and the use of wireless communication. Therefore, there are several possible optimizations to be made when designing solutions for WPT in EV.

Table 2.1 helps to get a perception of the time that a complete transaction may take, given the network communication distance and the vehicle speed. This time imposes restrictions on employed solutions such as the algorithms. This table does not differentiate range or radius of communication. There is actually the double of the time allowed to run the algorithm, but that margin is left for possible connection setups or retransmissions.

In the related studies, there are several works related to payment systems. For example, the author of [40], proposes an authentication algorithm called Trust-Extended Authentication Mechanism (TEAM), which is based on an hash function. It adopts the

Speed		*Wireless range (m)*				
(km/h)	*(m/s)*	*10*	*20*	*40*	*60*	*80*
160	*44,44*	0,23	0,45	0,90	1,35	1,80
120	*33,33*	0,30	0,60	1,20	1,80	2,40
100	*27,78*	0,36	0,72	1,44	2,16	2,88
80	*22,22*	0,45	0,90	1,80	2,70	3,60
60	*16,67*	0,60	1,20	2,40	3,60	4,80
40	*11,11*	0,90	1,80	3,60	5,40	7,20
20	*5,56*	1,80	3,60	7,20	10,80	14,40

Table 2.1: Time available to carry out a transaction (in seconds) given a vehicle speed and a range of the wireless network.

concept of transitive trusting relationships, where a vehicle becomes a trusted entity after authentication. The protocol, however, does not protect privacy, since the original identifier is shared during authentication.

There is also another solution [41], that offers two mutual authentication mechanisms between vehicles and the charging plates that can be used for different vehicular speeds on the road. One approach is based on direct authentication and the other on hash chain-based authentication. It also uses a game-theoretic approach, keeping an equilibrium during any step of the protocol.

There is another scheme based on Wireless Mesh Networks (WMN) [42]. It proposes a Secure Localized Authentication And Billing (SLAB) scheme for WMN to address the security requirements and performance efficiency, in terms of reducing inter-domain handoff authentication latency and computation load on the roaming broker.

Also based on wireless mesh networks, the UMTS Online Charging System (OCS) uses credits to pay for traffic volume or the duration time [43]. The work proposes a Credit Pre-reservation Mechanism (CPM) to avoid that credits are depleted at the Gateway GPRS Support Node (GGSN).

There is a work based on bank accounts, that specifically uses a hash-chain signed by a bank [44]. Since this entity is a trusted third party, it can manage users accounts. Zero knowledge proofs are used by a consumer to prove ownership of the hash-chain to a service provider.

In the work [45], it is suggested a local and proxy based authentication and billing scheme. It is used to reduce the additional cost of long distance communication overhead, due to the multi-hop forwarding in vehicular ad-hoc networks (VANET). The authors resort to a batch verification mechanism in their scheme to fulfill the security requirements and signature-based communications. Nevertheless, such scheme will not be ideal because of the large delay of the communication hops, therefore not being suitable for a fast energy transaction.

In [46] is proposed a platform that encompasses authentication, authorization and accounting (AAA) as a framework for purchasing services from Remote Service Unit (RSUs). It is used a signature-based and a key policy attribute-based encryption (KP-ABE) for the billing mechanism to attain localized fine-grained access control. It also employs a portable electronic currency.

Chapter 3

Low overhead protocol

This chapter describes one protocol that ensures privacy of meter readings by the electrical company, which is called Low Overhead Privacy protocol (LOPA) [47] [28]. In this protocol, the meters are organized in groups of a few hundreds. The meters collect their readings and send them to an aggregator, after applying some sophisticated cryptography. The aggregator collects the various readings, and by adding them, it gets the sum of all consumed energy. The aggregator might correspond to an electrical substation, and it also possess metering capability. Therefore, the aggregator can compare its local reading with the ones provided by the meters, and therefore detect (eventual) fraud. Within a group there is an initial key generation using the Diffie-Hellman algorithm (DH), and with that key, subsequent keys are made to encrypt the meter readings.

3.1 The LOPA protocol

In this protocol, the meters are organized in groups. Each meter has an unique m identification within the group. In a group there are j meters. A meter m performs its readings (V_m^i) in rounds identified by a number i.

The meters do a DH key generation with each other, creating a symmetric key between each pair. In other words, each meter has a shared symmetric key generated with every other meter in the group. Each of these keys are then hashed, generating $K_{m,k}$ that is shared between two meters m and k. In Figure 3.1 it is shown a simplified example of the keys shared between each pair of meters.

For each round i, each meter calculates $X_m^i = \sum_{k \neq m}^{j} (-1)^{k<m} h(K_{m,k} \,||\, i)$ (the $k < m$ part is a logical comparison, returning 1 if k is a smaller ID than m, or else it is returned

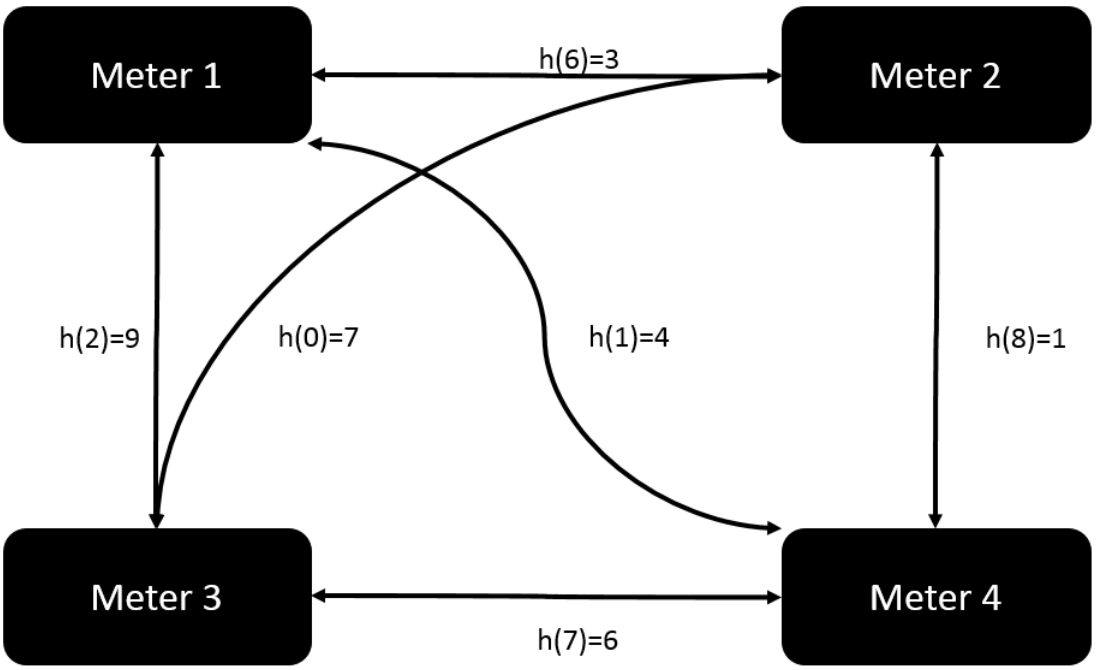

Figure 3.1: Example of a key for each meter pair among a group with four meters.

Meter 1	**Meter 2**	**Meter 3**	**Meter 4**
$K_{1,2} = 3$	$K_{2,1} = 3$	$K_{3,1} = 9$	$K_{4,1} = 4$
$K_{1,3} = 9$	$K_{2,3} = 7$	$K_{3,2} = 7$	$K_{4,2} = 1$
$K_{1,4} = 4$	$K_{2,4} = 1$	$K_{3,4} = 6$	$K_{4,3} = 6$

Table 3.1: Example of the list of $K_{m,k}$, that each meter stores (according to Figure 3.1).

the value 0). The value k is the unique identification of the other meters within the group, and recall that j is the number of meters in the group. That is, each meter calculates the hash of each $K_{m,k}$ concatenated with the round identifier i. Then, it subtracts all values that are from meters with a lower id number than itself, and sums all values that are from meters with higher id number than itself. The result is the creation of X_m^i. A simplified example can be observed in Tables 3.1 and 3.2. It is possible to see a simplified example of X_m^i being created within a group.

The meters sum their reading V_m^i with X_m^i ($C_m^i = V_m^i + X_m^i$) and send C_m^i to the aggregator. Next, the aggregator sums all C_m^i received from a group ($\sum C_m^i$) (see Table 3.2). Since the sum of the keys among a group adds to zero ($\sum X_m^i = 0$), this leads the aggregator to know exactly the sum of all readings ($\sum C_m^i = \sum V_m^i + X_m^i = \sum V_m^i$) and not the individual readings V_m^i. This provides the desired individual privacy and the needed global meter reading.

Meter ID	Intermediate calculations	Key generation (X_m^i)	Meter reading (V_m^i)	Encrypted reading (C_m^i)
1	+h(3$\|\|i$) + h(9$\|\|i$) + h(4$\|\|i$)	26	10	36
2	−h(3$\|\|i$) + h(7$\|\|i$) + h(1$\|\|i$)	8	24	32
3	−h(9$\|\|i$) − h(7$\|\|i$) + h(6$\|\|i$)	−15	0	−15
4	−h(4$\|\|i$) − h(1$\|\|i$) − h(6$\|\|i$)	−19	13	−6
	Sum ($\sum$)	0	47	47

Table 3.2: Simplified example of X_m^i key generation and encryption.

3.2 Implementation and evaluation

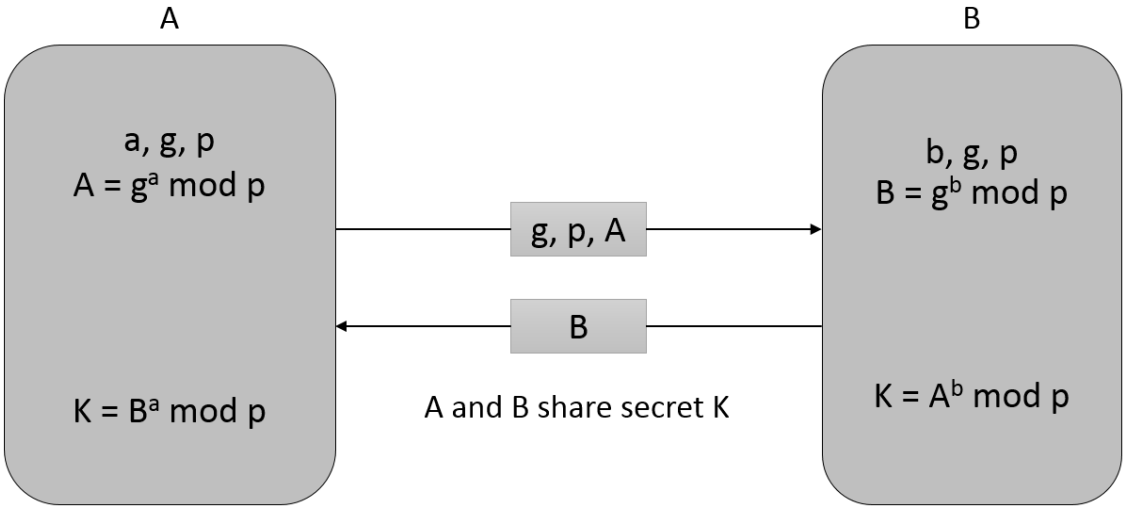

Figure 3.2: The general idea behind DH (inspired from [48]).

The DH key generation is, as the name suggests, an algorithm to generate a secret value between two or more parties, even if there are eavesdroppers listening to communications. In other words, the two parties exchange public data and create a shared secret from it, but the attacker cannot obtain the same secret. This secret value can then be used as a key for symmetric encryption. DH is based on a mathematical problem related to the difficulty of computing discrete logarithms. Its basic logic can easily be seen in

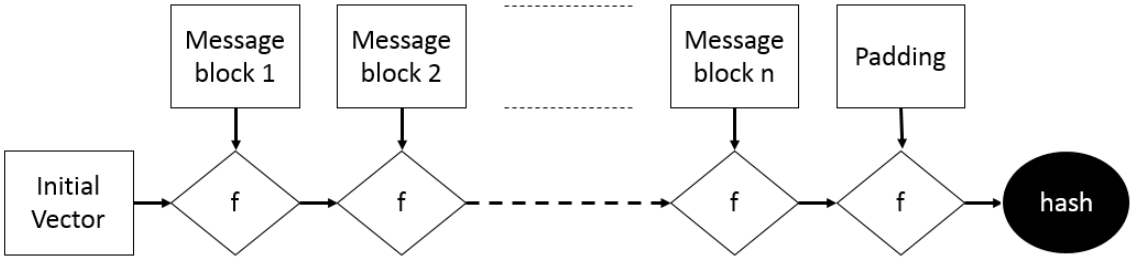

Figure 3.3: The structure of an hash function (from [49] [50]).

Figure 3.2. In our implementation of the LOPA protocol, we used an elliptic curve DH (ECDH). In particular we resort to the standard curve25519, where the parameters p and g are defined, $2^{255} - 19$ and 9 respectively (therefore the name curve25519).

The LOPA protocol also uses a hash algorithm. An hash algorithm receives a string as an input value and generates a fixed length result, the hash. An example of how an hash algorithm processes an arbitrary string is displayed in Figure 3.3. On a secure hash function it is infeasible to know the original text by observing the generated hash, or to find another equivalent string that produces the same hash (when two different strings produce the same hash it is called a collision). In out implementation of the privacy preserving protocol we employed the SHA256 hash algorithm.

Our implementation of LOPA was programmed in the C language. The DH key generation was built using the ECDH of curve25519 that was made available by [53].

The hash function was used from [54] [55]. This code was created with performance in mind, since it is oriented for bitcoin miners, where there is a financial incentive to be faster.

The key X_m^i and meter readings (V_m^i) are the same size, in this case, 32 bits. If the addition of X_m^i and V_m^i generates more than 32 bits(there is an overflow), only the least significant 32 bits are considered.

The created code was organized as shown in Figure 3.4. In addition, there are two other files, one with the curve25519 and the other with the SHA256. The first (curve25519-donna.c) has the function `int curve25519_donna( u8 *mypublic, const u8 *secret, const u8 *basepoint)` that is called in by function `void doit( unsigned char *ek, unsigned char *e, unsigned char *k)` in comm.c. The second file (sha256.c) has the function `void sha256( const unsigned char *message, unsigned int len, unsigned char *digest)` that is called by function `void hash(unsigned char h[], unsigned char v[])` from func.c.

The test bed was based on two settings: a PC with an Intel Core2Duo E6400 2.13

pa.c
+int main()

comm.c
+void timerBegin() +void timerEnd() +void otherInit1(int groupSize) +unsigned int nederivekey(int id, int groupsize,unsigned long long turn) +void initMax(int id, int groupsize)

func.c
+void imprimeTempo(struct timespec begin, struct timespec end) +void doit(unsigned char *ek, unsigned char *e, unsigned char *k) +void rgen(unsigned char *c,unsigned int size) +void hash(unsigned char h[], unsigned char v[]) +void concat(unsigned char result[], unsigned char key[], unsigned long long t) +unsigned int shortaaa(unsigned char in[BYTES])

Figure 3.4: The organization of the LOPA implementation.

GHz and a Raspberry Pi. The compiler that was used was the GCC version 4.6.3. The code was single threaded, not making use of the dual core CPU.

Various optimization flags were tried on the PC. The flags that yield better times are `-std=c99 -O3 -mtune=native -flto -ffast-math`. The execution time can be observed in Table 3.3 in the **PC** column. The protocol takes near to 0.388 millisecond for each curve25519 calculation, and takes under 0.001 millisecond for each X_m^i key generation per meter in a group.

Out prototype was also tested on an older Raspberry Pi, using the same code, but different compilation flags. The flags that yielded better results were `-std=c99 -flto -Os -ffast-math -funroll-loops`. The time obtained can be seen in Table 3.3 in the **Pi** column. The results show that it took 5.238 milliseconds per curve25519 calculation and 0.006 millisecond for each X_m^i key generation per meter in a group.

	Pi		**PC**	
Group size	$K_{m,k}$	$1000 \times X_m^i$	$K_{m,k}$	$1000 \times X_m^i$
20	198.862	124.928	14.926	14.384
40	410.092	247.379	30.600	29.528
60	615.016	383.054	46.106	44.655
80	843.432	493.796	61.898	59.884
100	1053.354	625.725	77.566	75.058
120	1250.912	767.928	93.238	90.208
140	1458.106	883.609	108.946	105.429
160	1680.446	1004.716	124.810	121.400
180	1905.260	1130.820	140.378	136.593
200	2093.294	1264.140	155.634	151.937
220	2320.812	1387.975	171.682	166.291
240	2522.838	1512.370	187.268	181.128
260	2738.284	1640.333	203.216	197.658
280	2953.450	1771.301	219.090	212.182
300	3163.156	1891.554	234.190	226.718
320	3368.270	2027.053	249.510	243.303
340	3642.200	2147.740	265.670	258.145
360	3828.930	2280.674	281.406	272.148
380	3996.480	2413.574	296.738	287.306
400	4203.740	2543.468	312.436	302.343

Table 3.3: Time in milliseconds that the algorithm takes to perform a step of the DH and 1000 X_m^i keys.

3.3 Summary of the findings

As expected, the results demonstrate that a weak device like Raspberry Pi (700 Mhz CPU) is slower than a regular computer (with 2.13 Ghz) when performing the cryptographic operations of the LOPA protocol. The main difference comes from the CPU architecture and compiler specific optimizations.

Nevertheless, the results of the Raspberry Pi still seem promising, since even with a large numbers of meters, it would be possible to complete the calculations within a reasonable time. Network delay will probably be the major factor to create delays, but it is mostly relevant during the initial $K_{m,k}$ key generation.

Chapter 4

Electrical vehicle energy purchase

This chapter proposes an architecture for a system that supports the payment of WPT to EV with batteries. It presents the main entities involved and the requirements, for example, the need for privacy and a limit for the duration of the protocol execution at a certain vehicle speed. Three energy payment protocols are explained, each one with different characteristics. Two of the protocols were implemented and the runtime values were obtained in two types of platforms. The results demonstrate that these protocols allow the completion of the transaction at normal driving speeds.

4.1 Payment system structure

To allow the payment of WPT to EV with batteries, a system structure needs to be defined. Some basic entities are easy to identify within the system, the EV and its user, the charging point and a regulatory entity.

4.1.1 System model

The proposed system includes the entities displayed in Figure 4.1. There is a trusted entity, which can be for example the Institute for Mobility and Transportation (IMT) or the Regulatory Authority for Energy Services (ERSE), which provides multiple pseudoanonymous profiles to the vehicles. This entity stores safely and privately the relationship between a pseudoanonymous profile and the corresponding vehicle, and this relationship is just released in the event of litigation.

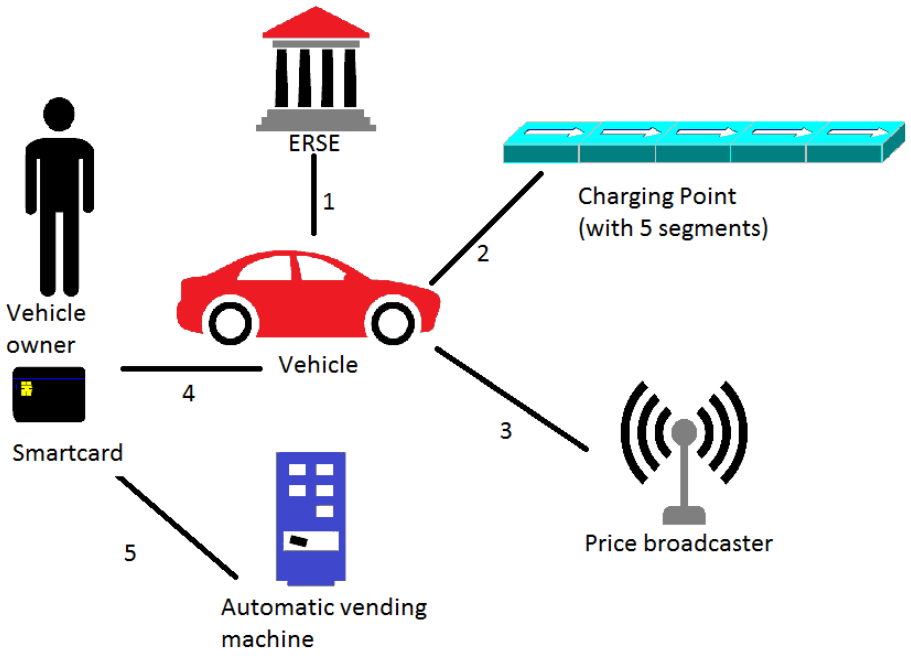

Figure 4.1: The system entities, including the main communications.

A charging point has a source of energy from the utility power infrastructure and a network connection to the automatic vending machines. In addition, it has the ability to process and communicate with the vehicle. The charging point is composed of segments. Therefore, the energy transfer is also segmented, i.e., each segment provides part of the total amount of energy that was requested.

Payment with traditional cash is performed by the person responsible for the vehicle, using a smart-card in an automatic vending machine or through an equivalent anonymous form of payment (e.g., Payshop or notes/coins). This person can be the lawful owner of the vehicle, a momentary user, a fleet manager or any other person who has responsibility for the vehicle.

The electric vehicle has energy transfer capacity and contains a simple device resistant to fraud, i.e., a Trusted Computing Base (TCB). If there is physical tampering with the device, this becomes evident. The device stores and uses the pseudoanonymous profiles or the digital money and/or generates an ephemeral anonymous identifiers (ID), according to the protocol implemented.

Both the vehicle and the charging point contain an energy meter that is also fraud resistant, i.e., a TCB. This TCB indicates the transferred energy and stores its history. In the case of vehicles, these functionalities can be included in the same TCB mentioned before.

4.1.2 Basis and assumptions

The certificates with the public keys of the trusted entities (ERSE or IMT) are securely installed in all charging points and vehicles (communication 1 in Figure 4.1). In the same sense that license plates are currently issued and installed before the vehicles can circulate, the IMT could supply the TCB devices that meet the requirements (i.e., resistant to fraud), and its installation can be carried out by the vehicle seller.

The communications between the vehicle and the charging point (communication 2 and 3 of Figure 4.1) may have low bandwidth, but should be reliable with a high probability, ensuring that the information reaches its destination. That is, a protocol (or a signal modulation) should be used in a way that favors reliability, for example by transmitting the same information at different frequencies or by adding error correcting coding.

The energy transfer is segmented [8], which can be employed to reduce the impact of theft or low efficiency in the energy delivery. This means that, if a problem occurs in one of the segments, the remaining transfers can be canceled, thus reducing the loss of energy compared to using only one segment that transmits the total agreed energy. This option could allow the utilization of simpler energy meters in the vehicles without being a TCB because, in case of loss, the wasted energy value is much lower. When implementing the system, there is a tradeoff to be made between lower cost of energy meters and the potential for small energy losses. It should be noted that there will always be a discrepancy between the emitted and received energy due to inefficiencies, and this should be taken into account in the design of the solutions.

The vehicles have a controller that makes the decision to buy energy, based on the travel plan (or the travel habits) and the options previously inserted by the vehicle driver. This controller takes into account the internal inefficiency, batteries wear rates, and the distance to other competing charging points. This last information, which includes the price of the energy and distance to charging points, is transmitted by the price broadcaster. These components are placed at various positions on the side of the roads. If the energy transfer inefficiency becomes the responsibility of the vehicle, the controller also takes into account that risk when making the decision to charge.

There must be vending machines that support anonymous transactions, allowing for example cash payments. These machines must be independent from energy sellers because a user will normally employ the closest one to home (or work) for convenience. It should not be possible to correlate a person to a payment for energy transfer even if that person always resorts to the same machine. The generated information (ID and password) is sent in a convenient but safe way, for example by storing it in a smart card (or other device like a USB token) provided by the person. Communications between the vending machine and the vehicle (communication 4 and 5 of Figure 4.1) can therefore

be accomplished with a smart-card that transmits information by physical contact or by some alternative means, such as Near Field Communications (NFC).

The user uses its smart-card (communication 4 in Figure 4.1) by inserting it into the vehicle, and after authentication, for example with an access pin, the vehicle will receive digital money or one (or more) account identifier and its password. This password prevents the use of account identifiers by malicious attackers that try random account numbers for the purpose of misusing the balance of the prepaid accounts or using the stored information in case of a lost smart-card.

The vehicle and charging point keep reports of the payment transactions during a period of time. This interval can be defined in terms of months, weeks or days (in case of the pseudoanonymous protocol). However, since messages have a reduced size, this leads to a low information storage cost.

A vehicle that is potentially interested in acquiring energy (communication 3 in Figure 4.1) will listen for messages from the price broadcaster. After validating those messages, the vehicle analyses the prices, and if it wants to charge the battery, it may precalculate the messages to be sent and drive over the charging segments.

4.1.3 Price Broadcaster

All charging points have one (or more) price broadcaster located at some distance before a vehicle reaches the charging segments. This broadcaster emits only one-way (communication 3 in Figure 4.1), i.e., there are no responses sent back by the vehicle. The broadcast contains information about energy prices in the next charging points. Maintenance of the price broadcaster is a responsibility of the charging points.

Energy prices are continuously issued with wireless communication. This information should be sent so that the vehicles receive it before they reach the transfer segments, and have time to process it beforehand.

The data broadcasted should contain rules and price lists for the energy purchase, the span of validity of the prices, the location of charging points in which the tariff is valid, and the current date (so it can be used as a proof of validity). The message contains a digital signature of its content. It also includes two digital certificates issued by a trusted entity (e.g., ERSE or IMT), where one public key is for encryption (the $Kupc$ key) and the other to validate digital signatures (the $Kups$ key). The transmitted message is represented in Table 4.1. Such a message may contain information related to more than one charging points.

When a vehicle receives a price broadcast message, it validates the certificates by

(PointID $\|\|$ "Price table: 1kw at 0,1euro during 8pm-6am Jun2015, ..." $\|\|$ "Local: x,y,z. 3 segments" $\|\|$ date)$_{Krps}$ $\|\|$ certificate with $Kups$ $\|\|$ certificate with $Kupc$

Table 4.1: Message content issued with the pricing of a charging point. Legend: $(\ )_{Krps}$ content signed with the $Krps$ key. This signature is validated with the public key $Kups$ that is included in a certificate.

checking if they are digitally signed with a private key of a trusted entity (ERSE or IMT). Then, it verifies if the message signature is valid using the public key $Kups$ on the certificate, and finally checks the date. The properties of integrity and authentication of charging points are assured by the signature. The charging point identifier and the certificates can be stored by the EV, thus avoiding repeated validations in the future.

4.2 Payment Protocols

This section presents the details of each of the three protocols. The protocols comply with the requirements and features mentioned before, and they have a common basis of resorting to the same kind of entities like the vending machines or the price broadcaster. These protocols also have a similar message structure, but there are some differences between them. The first protocol, called Pseudoanonymous Profiles, performs the payment after the energy transfer, and requires the trusted entity to intervene more (creating the profiles and securing their connection with the vehicle) than the other protocols. The two other protocols are prepaid. The Anonymous Pre-Payment is based on accounts that have a balance and the Anonymous Digital Money protocol uses digital money, which has the ability to work even if the charging point has no connection to the vending machines network, during the payment transaction.

To describe the content of the messages of each protocol, the following notation is used. The digital signature keys employed are: $Krts$ and $Kuts$ are respectively the private and public key of the vehicle's TCB; $Krvs$ and $Kuvs$ are respectively the private and public key of the vehicle; $Krns$ and $Kuns$ are respectively the private and public key of the payment network; and finally $Krps$ and $Kups$ are respectively the private and public key of the charging point. $Krpc$ and $Kupc$ keys are respectively the private and public key of the charging point for encryption operations; also $Krnc$ and $Kunc$ are respectively the private and public encryption keys of the payment network. A character set is within quotation marks " ", the concatenation operation is represented by $||$, and an optional field lies within $<\ >$. When there is $E_{key}[\ ]$, it means that the content between square brackets is encrypted with the key key. Information that is digitally signed, for example

with the key key, appears in brackets: $(\ )key$. After the identifier for each message, there is the symbol $V \Rightarrow P$ or $P \Rightarrow V$, indicating the message's direction, respectively from the vehicle to a charging point or from a charging point to the vehicle.

4.2.1 PP – Pseudoanonymous Profiles

1	$V \Rightarrow P$	(1 || PseudoID || PointID || E_{Kupc}[sym] || E_{sym}[IDtrans || date || "buy 1000W at 0,1 euro"])$_{Krts}$ || TCB certificate with $Kuts$
2	$P \Rightarrow V$	(2 || PointID || PseudoID || E_{sym}[IDtrans || "OK 100+400+500" || Invoice])$_{Krps}$
3	$V \Rightarrow P$	(3 || PseudoID || PointID || E_{sym}[IDtrans || "OK"])$_{Krts}$
4	$P \Rightarrow V$	*Transfer 100W of energy*
5	$V \Rightarrow P$	(4 || PseudoID || PointID || E_{sym}[IDtrans || "OK" || "TCB:78W"])$_{Krts}$
6	$P \Rightarrow V$	*Transfer 400W of energy*
7	$V \Rightarrow P$	(5 || PseudoID || PointID || E_{sym}[IDtrans || "OK" || "TCB:378W"])$_{Krts}$
8	$P \Rightarrow V$	*Transfer 500W of energy*
9	$V \Rightarrow P$	(6 || PseudoID || PointID || E_{sym}[IDtrans || "OK" || "TCB:478W"])$_{Krts}$

Table 4.2: Example of a complete transaction using the pseudoanonymous profiles protocol.

The protocol based on the pseudoanonymous profiles requires a trusted third party entity, for example, the regulatory authority of energy or transport (e.g., IMT or ERSE). This entity creates multiple profiles for each vehicle TCB, securely storing the association between vehicle (or owner) and the multiple profiles. This association is revealed only in the event of non-payment by the vehicle's owner, typically after a request from a court of law.

The vehicle's TCB randomly uses a profile when it needs to make a transaction, and

the more profiles available to the vehicle's TCB, the less is the probability of someone making a correlation to violate privacy. Ideally, one different profile should be utilized per purchase.

An anonymous profile contains an identification number (`PseudoID`) and a pair of asymmetric keys for digital message signatures ($Krts$ and $Kuts$). Furthermore, it includes a certificate containing the public key $Kuts$, which is signed by the trusted entity (ERSE or IMT). These profiles are inserted into the vehicle's TCB with a smart-card (or other device like a USB token) before any transaction can be performed.

The transaction begins when a vehicle receives a message from a price broadcaster, which indicates the entire pricing list (see Table 4.1). The broadcast message is signed and contains two certificates.

When a vehicle is interested in an energy transfer, it initiates a communication (message 1 in Table 4.2), using one of its pseudoanonymous profiles selected randomly. The message states the intention to buy a certain energy quantity, confirming the price and signing the message. The message is partially encrypted, with a randomly generated key sym. This key is then encrypted with the certified public key of the charging point that was previously known through the price broadcast.

The charging point validates all the information, including the signature's validity, and if the certificates are issued by the trusted entity and whether such certificates are not included in a revocation list. It is also checked if the vehicle's intention to buy a certain amount of energy can be satisfied. In addition, it verifies if the selected choice is included in the price list and whether the transaction number is new in the current date. After that, it responds confirming the possibility of energy transfer (message 2). The message indicates the characteristics of the charging point, such as the number of segments (there are 3 segments in the example of Table 4.2) in the energy transfer zone or the amount of energy that can be transferred. It also includes the invoice, containing the same transaction number.

In the third message, the vehicle confirms the charging point's information, indicating that it accepts the proposed conditions.

After three correctly exchanged messages, the energy transfer can occur (the energy exchange itself is identified as messages 4, 6 and 8). Each transfer is confirmed with an acknowledgment message from the vehicle (message 5, 7 and 9), indicating the actually received amount of energy. If one of these energy transfers is insufficient, instead of an `"OK"` message, it is sent a cancellation message. Later on, it should be examined whether there are defects in the charging point or in the vehicle.

After the transaction, the charging point sends the invoice identifier and the invoice

itself to the payment network, which corresponds to the last encrypted field of the message 2 of Table 4.2.

Subsequently, the vehicle stores the invoice identifiers, e.g., in a smart-card, which can be used later to pay the bills (being the cash payment done, for example, in an automatic vending machine, or in any other equivalent form of payment).

If an invoice is not paid in a pre-determined time, the charging point can report the problem, presenting the exchanged messages as evidence. The trusted entity reveals the real identity of the vehicle's owner, and revokes all certificates issued to the vehicle. This will prevent the vehicle from being charged in the future until the problem is settled.

4.2.2 APP – Anonymous Pre-Payment

1	$V \Rightarrow P$	(1 || EmitID || PointID || E_{Kupc}[sym] || E_{symTCB}[(Krvs)$_{Krts}$] || E_{sym}[$Kuvs$ || IDtrans || EmitID || date || "buy 1000W @ 0,1euro" || AccID || passAcc])$_{Krvs}$
2	$P \Rightarrow V$	(2 || PointID || EmitID || E_{sym}[IDtrans || "OK:100+400+500"])$_{Krps}$
3	$V \Rightarrow P$	(3 || EmitID || PointID || E_{sym}[IDtrans || "OK"])$_{Krvs}$
4	$P \Rightarrow V$	*Transfer 100W of energy*
5	$V \Rightarrow P$	(4 || EmitID || PointID || E_{sym}[IDtrans || "OK" || "TCB:78W"])$_{Krvs}$
6	$P \Rightarrow V$	*Transfer 400W of energy*
7	$V \Rightarrow P$	(5 || EmitID || PointID || E_{sym}[IDtrans || "OK" || "TCB:378W"])$_{Krvs}$
8	$P \Rightarrow V$	*Transfer 500W of energy*
9	$V \Rightarrow P$	(6 || EmitID || PointID || E_{sym}[IDtrans || "OK" || "TCB:478W"])$_{Krvs}$

Table 4.3: Example of a complete transaction using the anonymous pre-payment protocol.

In the anonymous pre-payment protocol, the vehicle's owner makes a payment anonymously, for example in a vending machine that accepts cash. In exchange, the owner

("Transaction 1" ‖ date ‖ EmitID ‖ PointID ‖ sym ‖ symTCB ‖ Krvs ‖ "buy 1000W @ 0,1euro" ‖ AccID ‖ passAcc ‖ (2 ‖ PointID ‖ EmitID ‖ E_{sym}[IDtrans ‖ "OK:100+400+500"])$_{Krps}$ ‖ 1 ‖ "TCB:0W" ‖
"Transaction 2" ‖ ...)$_{Krts}$ ‖ TCB certificate with $Kuts$

Table 4.4: A message from the TCB disclosing the transactions and demonstrating the responsibility of the transmitted messages.

receives in his smart-card one or more identifiers of temporary prepaid accounts and their access keys (e.g., an account password). The user can then insert the smart-card in the vehicle and authenticate, for example with an access pin, and then the smart-card allows the vehicle to use the prepaid accounts. In this protocol, a TCB is required to be installed on vehicles. As usual, it is assumed that the TCB is correct and that no user can produce a modified malicious TCB.

The charging point requires connection to the payment system network to validate the balance of the temporary prepaid account. This network includes the automatic vending machines and a central server (or something equivalent) that maintains records of the prepaid accounts.

The main differences of this protocol over the previous one is the addition of the temporary account identifier (`AccID`) and its password (`passAcc`) in the first message, and the elimination of the invoice from the second message (see Table 4.3).

An ephemeral identifier (`EmitID` and a key pair, the $Kuvs$ and $Krvs$) is used to identify the vehicle during a transaction in the first message (see Table 4.3). A new identifier is created by the TCB for each transaction. The first message also has a public asymmetric $Kuvs$ key, which serves for the charging point to verify the digital signature of the messages, protecting the integrity of the data sent by the vehicle.

After receiving the first message, the charging point validates all the information, including the signature's validity. It also checks if the vehicle's intention to buy a certain amount of energy can be satisfied. In addition, it verifies if the selected choice is included in the price list and if the transaction number is new in the current date. Then, the charging point verifies with the payment network if the given account number and access password are valid and have enough balance for the transaction.

After that, it responds confirming the possibility of energy transfer (message 2). The message indicates the characteristics of the charging point, such as the number of segments (there are three segments in the example of Table 4.3) in the energy transfer zone or the amount of energy that can be transferred.

After three correctly exchanged messages, the energy transfer can occur (the energy exchange itself is identified as messages 4, 6 and 8). Each transfer is confirmed with an acknowledgment message from the vehicle (message 5, 7 and 9), indicating the actually received amount of energy. If one of these energy transfers is insufficient, instead of an "OK" message, it is sent a cancellation message. Later on, it should be examined whether there are defects in the charging point or in the vehicle.

In this protocol, the charging point only allows the transaction to proceed after checking that it can get the funds stored in the temporary account. Therefore, it will always be payed, for instance, by requesting a transfer of money from the temporary account to its own account before transmitting message number 2 of Table 4.3.

The protocol also ensures that a misbehaving charging point cannot steal money from the user without being discovered. The first message of Table 4.3 includes a field ($\mathtt{E}_{symTCB}\mathtt{[(Krvs)}_{Krts}\mathtt{]}$) that allows a user to demonstrate to a third party (e.g., a judge) that he was the rightful owner of the temporary account. If necessary, the user can request a proof of ownership from the car TCB (see Table 4.4); where the key $symTCB$ is made available, together with other information from the transaction. This key allows the third party to confirm that $Krvs$ was generated by the TCB with private key $Krts$, which is associated with the car of a particular user. Since this key $Krvs$ was used to sign message 1 of Table 4.3, then the account belongs to the user.

4.2.3 Generation and security of digital notes

This section explains the main features of the digital note used by the Anonymous Digital Money protocol. This solution is inspired on the works of [56] and [57]. The vehicle will generate non-valid digital notes when the processor is idling. All individual notes have the same value v, different serial numbers, and the same predetermined validity interval. It then generates a list of size p with the owner's real identity (for example, using as identifier the driving license number), divided into two segments. The vehicle also generates $2p$ symmetric keys and encrypts the identity list, as shown in Table 4.5 and 4.6.

The notes are multiplied by a value called the blinding factor, and then are signed by the owner TCB (for example, with the smart-card certificate). Next, each of the notes is encrypted with the public key of the payment system network, $Kunc$. The blinding factor is used to prevent the payment network to know exactly which note it will generate (e.g., through the serial number), allowing for an anonymous note to be created. The vehicle stores in a secure smart-card the various notes, the keys to decrypt the identification lists and the blinding factors.

When the vehicle's owner goes to an anonymous vending machine with his smart-

Owner real identity: 1234567890
A random number 1: 5645684166
A random number 2: 3453463455
...
A random number p: 1241780703
$E_{symLef1}$["Left" $\|$ 5645684166] $\|$
$E_{symRig1}$["Right" $\|$ (5645684166 $\oplus$ 1234567890)]
$E_{symLef2}$["Left" $\|$ 3453463455] $\|$
$E_{symRig2}$["Right" $\|$ (3453463455 $\oplus$ 1234567890)]
...
$E_{symLefp}$["Left" $\|$ 1241780703] $\|$
$E_{symRigp}$["Right" $\|$ (1241780703 $\oplus$ 1234567890)]

Table 4.5: Example of the identification list. Legend: E_{symLef}[ ] indicates the encryption with $symLef$ symmetric key; a similar action occurs with E_{symRig}[ ]; $\oplus$ is a XOR operation.

card, he delivers n digital notes with the same value. In return, the machine requires $n-1$ blinding factors and all the symmetric keys for the $n-1$ identification lists. Then, the machine checks the validity of the $n-1$ notes by decrypting all identification lists, and verifying that all have the same owners identity and that they correspond to the driving license number (which can be in the smart-card). If every note is valid, the payment network signs the remaining note, which still has the blinding factor and keys without being disclosed. Then, it returns the note signature to the smart-card. Next, the smart-card erases all revealed notes, their keys and blinding factors. The remaining note becomes valid for payments after being transferred to the vehicle, and removing the blinding factor from the signature. The resulting signature corresponds to the unblinded note (see Table 4.6).

There is a probability $\frac{1}{n}$ of the vehicle's owner to be able to create an erroneous note

(serial number $\|\|$ validity period $\|\|$ value $\|\|$
$E_{symLef1}$["Left" $\|\|$ R_1] $\|\|$ $E_{symRig1}$["Right" $\|\|$ $R_1 \oplus$ ID] $\|\|$
$E_{symLef2}$["Left" $\|\|$ R_2] $\|\|$ $E_{symRig2}$["Right" $\|\|$ $R_2 \oplus$ ID] $\|\|$
... $\|\|$
$E_{symLefp}$["Left" $\|\|R_p$] $\|\|$ $E_{symRigp}$["Right" $\|\|$ $R_p \oplus$ ID] $)_{Krns}$

Table 4.6: The content of a digital money bill. Legend: E_{sym}[ ] encrypted with sym symmetric key; ($)_{Krns}$ signed with $Krns$ key from the payment machines network.

with another value without being discovered. However, if detected with an erroneous note, various actions can be taken, such as preventing the person from acquiring further notes with a temporal or permanent ban. Moreover, the value of n can be increased to ensure that future attempts of fraud will be discovered with a higher probability. The notes may also be limited to a maximum value, for example 10 euros, which may be the maximum value for a transaction, thereby also minimizing the risk.

The serial number allows the detection of multiple spending, where the same note is used more than once. This is achieved through the payment network that stores spent notes during their validity period, and checks if a note is already spent when it is being used for a payment. However, if the charging point does not have enough connectivity to the anonymous payment network, it asks the vehicle for the keys to decipher half of the identification list, which allows to determine which entity is guilty in case of duplicate use of a note (either the vehicle owner or the charging point). The selection list is a binary string of size p, corresponding to the Left or Right segment of the identification list (see Table 4.5 and 4.6). The vehicle responds with the corresponding symmetric keys. Then, the charging point decrypts them and checks that the decrypted part is valid by verifying that the text begins either with "Left" or "Right". The half of the identification keys that the charging point possesses is also stored along the serial number.

If the same note is submitted to the payment network (same serial number and with equal half identification list decrypted), the charging point is very likely to be the guilty party because there is only $\frac{1}{2^p}$ probability of two charging points asking the same binary string to the user. If the same note is submitted to the payment network (same serial number but with two different half identification lists decrypted), then the fault probably is from the owner of the vehicle. In the later case, it is selected two halves of the same index of the selection list, and an exclusive or (XOR) operation is done with the two halves, creating the original identification of the owner of the note. This probability is parametric, and can be defined as high as wished, being a tradeoff between note storage

size and higher security.

4.2.4 ADD – Anonymous Digital Money

This protocol requires that the charging points and the vehicles have the certificates from the payment system network, so the vehicle can create the blinding factor and the charging point can validate the digital signature of the note.

The protocol that uses digital money is similar to the previous protocols (see Table 4.7), except that it includes one (or more) digital note(s) with the transaction value in the first message.

Like the previous protocol, an ephemeral identifier is created by the TCB to be used to identify the vehicle during a transaction (`EmitID` and a key pair, the $Kuvs$ and $Krvs$), which is given to the charging point in the first message. The first message also has the public asymmetric $Kuvs$ key, which serves for the charging point to verify the digital signature of the messages, protecting the integrity of all messages sent by the vehicle.

In addition, like in the previous protocol, after receiving the first message, the charging point validates all the information, including the signature's validity. It is also checks if the vehicle's intention to buy a certain amount of energy can be satisfied. In addition, it verifies if the selected choice is included in the price list and if the transaction number is new.

Then, it checks with the payment system network if the note(s) serial number has already been registered. If all checks are valid, the note can be accepted, and its serial number is stored as spent.

There are two messages that have additional fields, namely the second and third messages. The first extra field is in the second message, which is sent by the charging point. It sends a random binary string of size p. In the third message, the vehicle adds the symmetric keys that can decrypt the selected owner identification, consisting of a list of the symmetric keys to decipher the left or right parts (accordingly to the value of the selection list).

If there is an interruption in the transaction, it should be possible to do a partial or total refund. The way to achieve this is for the charging point to return a digital note to the vehicle. The vehicle then answers with a confirmation message. The charging point should have a set of valid notes to be used in case of refund. A final message containing a password to this digital note can be added, forcing the vehicle to confirm the receipt of the note.

1	$V \Rightarrow P$	(1 || EmitID || PointID || E_{Kupc}[sym] || E_{symTCB}[(Krvs)$_{Krts}$] || E_{sym}[$Kuvs$ || IDtrans || date || "buy 1000W@0,1 euro" || digital bill:0,1euro])$_{Krvs}$
2	$P \Rightarrow V$	(2 || PointID || EmitID || E_{sym}[IDtrans || "OK:100+400+500" || selection list])$_{Krps}$
3	$V \Rightarrow P$	(3 || EmitID || PointID || E_{sym}[IDtrans || "OK" || IDs])$_{Krvs}$
4	$P \Rightarrow V$	*Transfer 100W of energy*
5	$V \Rightarrow P$	(4 || EmitID || PointID || E_{sym}[IDtrans || "OK" || "TCB: 78W"])$_{Krvs}$
6	$P \Rightarrow V$	*Transfer 400W of energy*
7	$V \Rightarrow P$	(5 || EmitID || PointID || E_{sym}[IDtrans || "OK" || "TCB: 378W"])$_{Krvs}$
8	$P \Rightarrow V$	*Transfer 500W of energy*
9	$V \Rightarrow P$	(6 || EmitID || PointID || E_{sym}[IDtrans || "OK" || "TCB: 478W"])$_{Krvs}$

Table 4.7: Example of a complete transaction using digital money protocol.

4.3 Implementation

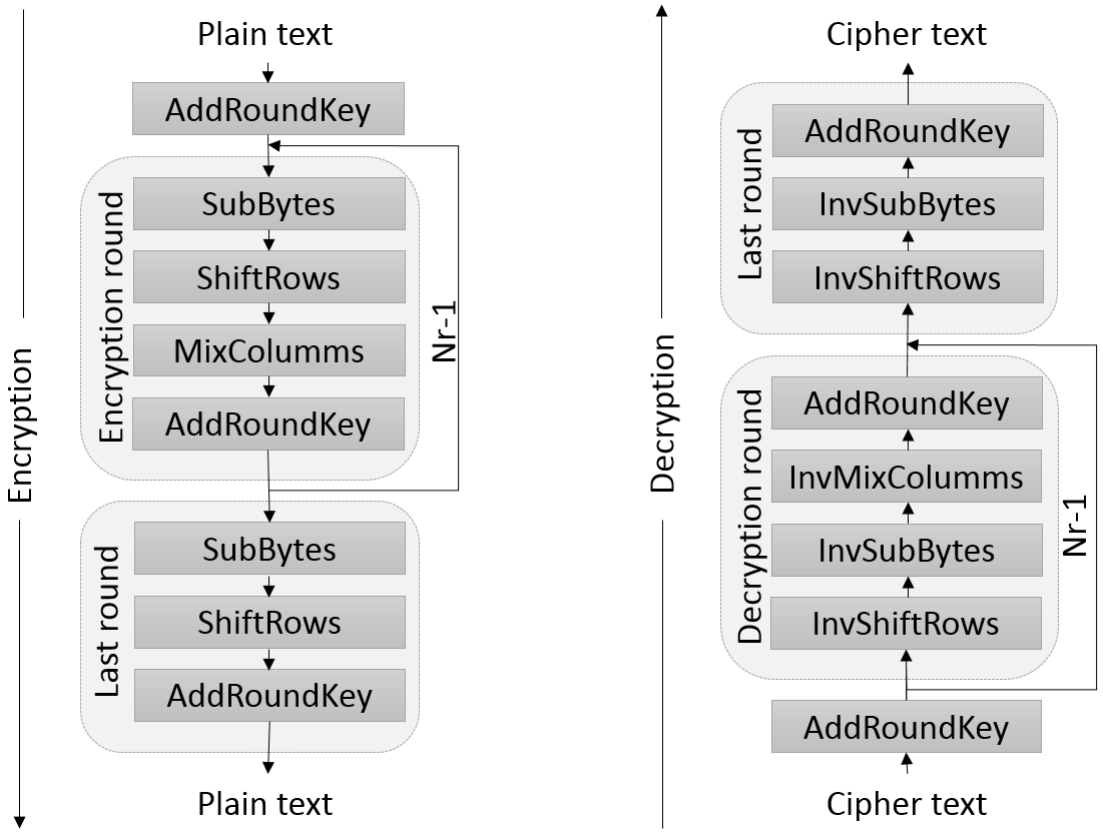

Figure 4.2: General structure of the AES algorithm (inspired from [58], [59] and [60]).

The three main cryptographic algorithm that are employed are the SHA, AES and RSA. SHA is an hash algorithm, as shown before in Section 3.1, but this time it was used the Java native implementation. AES is a symmetric encryption algorithm that uses the same key to encrypt or decrypt information. In other words, the sender and receiver must have the same key to exchange confidential information. AES divides the information in equal sized segments, called blocks. Then, it performs various rounds of operations on the block together with the key, to generate the encrypted block. To decrypt a block, the inverse process is carried out. The general structure of the AES algorithm can be seen in Figure 4.2.

RSA is an asymmetric encryption algorithm, and it uses two keys, a public (known by everybody) for encryption and a private key (only known by the owner) for decryption. An asymmetric encryption algorithm allows encrypted information to be sent without a prior key exchange, but it is generally slower than symmetric encryption algorithms. So in practice, RSA is usually used to transmit an encrypted symmetric key, and the actual content is then encrypted with the symmetric algorithm. The main calculations carried out by RSA can be seen on Figure 4.8.

Key Generation	
Select p, q	p, q both prime, $p \neq q$
Calculate $n = p * q$	
Calculate $\phi(n) = (p-1) * (q-1)$	
Select integer e	$g.c.d.(\phi(n), e) = 1;\ 1<e<\phi(n)$
Calculate d	$d \equiv e^{-1}\ mod\ \phi(n)$
Public key:	$Ku = \{e, n\}$
Private key:	$Kr = \{d, n\}$

Encryption	
Plain text (M)	$M<n$
Encrypted text:	$C = M^e (mod\ n)$

Decryption	
Encrypted text (C)	C
Plain text:	$M = C^d (mod\ n)$

Table 4.8: The main calculations of the RSA algorithm (inspired in [61] and [62]).

The RSA algorithm can also be used to create digital signatures, in combination with an hash algorithm. When someone wants to make a digital signature of some information, first he or she does an hash of that information, and then encrypts that hash value with the private key. To validate the signature, the verifier decrypts the signature with the public key, getting the hash of the signed information. Then he or she does an hash of the information, and compares both hash values. If both are the same value, then the signature is valid.

To make the digital money possible with all required properties, blinded signatures are used. To do so, first a blinding factor is generated r. It must be relatively prime to n (i.e., $gcd(r, n) = 1$). Then, the digital note m is multiplied by r and exponentiated with e. The result is the creation of a blinded note m'. Next, m' is sent to the bank in a secure way (e.g., signed by the user). After the bank has signed, it returns the blind note to the user. The user multiplies s' with the inverse of r, generating s a signature valid for m. This is represented in Figure 4.3.

The payment protocol based on digital money has been implemented in Java, using the bouncycastle library to calculate the blinding factor and RSA operations. For the symmetric cipher, the standard AES with a 128 bits key was used, and for the asymmetric

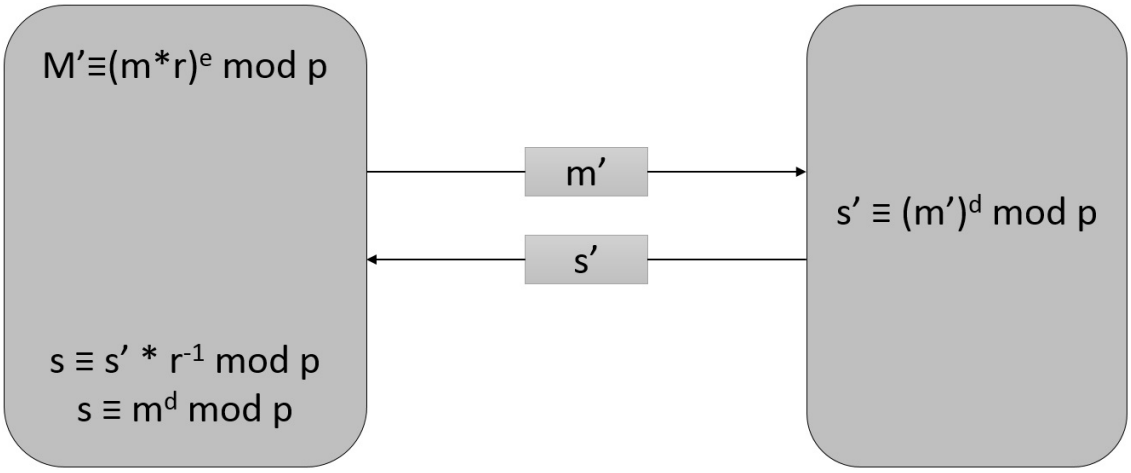

Figure 4.3: The main calculations of a blind RSA signature (inspired from [61] and [63]).

cipher, the RSA algorithm with 2048 bits keys was employed. Digital signatures are based on the SHA256 hash function and RSA with 2048 bits keys. The selection list of the notes has a size of 30 entries.

The structure a note can be seen in Table 4.6. Taking into account the size of the keys (2048 bits for asymmetric keys and 128 bits for symmetric keys) and of the selection list (30 entries), a digital note occupies around 1228 bytes plus 960 bytes for the passwords to access the selection lists. A megabyte smart-card can be used to save four hundred notes, and these smart-cards are already mass produced.

The pseudoanonymous profiles protocol was also implemented, following similar approaches for the cryptographic operations. The pre-payment protocol was not implemented because from a performance perspective it does not differ significantly from the other two. The construction of messages and their verification is similar, which are the steps that require the most processing.

It is possible to implement some optimizations on our current prototype, such as using encryption and digital signatures based on elliptic curves. The messages can be precalculated before they need to be sent, and verifications can be done in parallel (except message 3 on 4.7). Since these enhancements have not been implemented, there is room for some improvement.

4.3.1 Evaluation

The implementation was tested in two platforms, one based on a Raspberry Pi 1 and another on a Linux computer with an Intel Core2Duo E6400 2,13Ghz processor. Each experiment was run 1000 times and the mean result values are presented in milliseconds.

The summary of results is displayed in Tables 4.9 and 4.10.

These two different types of hardware were chosen to run the tests because they give an indication (below and above) of the performance of a real system. The implementation of the controller in the vehicle will probably be based on a hardware with a higher processing capacity than the Raspberry Pi 1, which is an outdated equipment, but hardly with a higher capacity than a computer with the CPU Intel E6400.

#	**Task**	**PC**	**Pi**
1	Generate selection list:	0,12	3,11
2	Generation of not yet valid note:	22,69	892,77
3	Sign note with blinding factor:	12,87	693,58
4	Remove the blinding factor:	1,54	48,90
5	Decrypt half of the identification:	0,08	14,50
Total		37,3	1652,86

Table 4.9: Time in milliseconds for different steps related to the creation and validation of the digital notes. PC and Pi correspond to the platforms with the computer PC or the Raspberry Pi respectively.

Tables 4.9 and 4.10 contain a set of numbered tasks. The time it took to run some operations was significantly longer due to the fact that they require complex calculations. The most computationally heavier tasks are the ones using the RSA algorithm, as in the case of digital signatures and asymmetric encryption (exponentiation with the private key is much slower than the one with the public key). Therefore, analyzing the tasks by the complexity of the calculations gives a good sense of what determines most of the runtime delays. Starting with the verification of digital signatures, these occur in tasks 3 and 4 of Table 4.9 and tasks 2, 4 (doubly), 6, 8 and 10 of Table 4.10. The task with the greatest computational burden is the creation of digital signatures, which is carried out in tasks 1, 3, 5, 7 and 9 of Table 4.10 and task 2 of Table 4.9. Finally, the last operation that uses the RSA algorithm is the encryption of data in task 3 and its decryption in task 4 of Table 4.10. There is also an encryption or decryption operation in all tasks, but in some cases they are held multiple times in tasks 2 and 5 of Table 4.9.

The generation of selection lists (task 1 of Table 4.9) depends on the capability of the operating system and hardware to provide good quality random numbers. However, the observed generation speed is more than sufficient for the needs.

#	Task	PC	Pi
1	Create broadcast message:	12,38	662,78
2	Verify broadcast message:	1,01	61,81
3	Create message 1:	13,59	697,08
4	Verify message 1:	13,46	723,08
5	Create message 2:	12,50	662,71
6	Verify message 2:	0,52	31,70
7	Create message 3:	12,60	661,20
8	Verify message 3:	0,52	31,72
9	Create confirmation message:	12,56	661,04
10	Verify confirmation message:	0,53	31,95
Total		79,67	4225,07

Table 4.10: Time in milliseconds for each step of the PP protocol. PC and Pi correspond to the platforms with the computer PC or the Raspberry Pi respectively.

For an energy transaction in a moving vehicle, the most important times are the ones corresponding to tasks 4, 6 and 8 of the Table 4.10. This happens because some messages can be created before the vehicle reaches the charging point (e.g., message 1), after receiving a broadcast with prices and having decided to buy energy. These tasks combined correspond to approximately 786 milliseconds with the Raspberry Pi. These delays seem acceptable even with a communication latency of 50 milliseconds added three times. Recall that with a range of 40 meters and a speed of 120 km/h, it takes 1,2 seconds to go through that distance (Table 2.1). In the case of the digital money protocol, task 5 (Table 4.9) is also important (adding 14.5 milliseconds), but the protocol is still functional with a range of 40 meters for a vehicle moving at 120 km/h.

The distance between the price broadcaster and the charging point must be sufficient to allow the execution of tasks 2, 3, 5, 7 and 9 of Table 4.10 before reaching the charging point. This corresponds to 2743 milliseconds all combined. Therefore, the price broadcaster must be at least 95 meters from the charging point. If tasks are parallelized (except in ADD protocol when selection strings are used) then 25 meters is enough.

Chapter 5

Conclusions

This project addressed two scenarios related to the energy transfer in a smart grid setting, while keeping the privacy of the users. The first scenario was related to the measurement of energy consumption with smart meters while the later is about payment protocols for wireless power transfer of electric vehicles in movement.

Smart meters will be part of our future, as they are currently being deployed in several countries. They can be used in several situations. One of them is consumption peaks, when consumers use more energy than usual. To provide enough energy during these peaks, energy generation needs to be increased to be able to cope with this high demand. But if these peaks could be reduced (or avoided) there would not be a need of such spare capacity of energy production, and therefore less money would be spent.

Another situation is the micro and mini production of energy. As the prices of small renewable energy generation get lower, more consumers are also producing energy. When they produce in excess, that energy should go to the grid; but that requires coordination and control between the electrical companies and the users. When there is not enough production to meet demand, this creates problems like a black out. With smart meters, those can be avoided by limiting the flow of energy consumed, in order to prioritize energy (e.g., to hospitals) and maintain a good quality service (even if somehow limited).

There is also another advantage related to smart meters, which is automatic meter readings. This avoids the need to have a person physically inspecting all meters, in order to collect readings to produce bills. In addition, it supports the remote enabling or disabling of the meter.

Payment protocols for WPT on EV have some requirements like speed of execution, security, bill the received energy, and also provide privacy to the user. This ensures that WPT technology can be used with the same privacy as a traditional fuel purchase,

providing similar guarantees as when paying with cash.

It was selected the LOPA protocol for anonymous meter readings. This protocol supports the secure transmission of the readings from a group of meters. A special method for key generation was employed, allowing individual readings to be encrypted with different keys. The sum of all (encrypted) readings yield the (not encrypted) total sum. The protocol was implemented and evaluated showing interesting results.

The structure for the payment system of WPT was laid, including the necessary entities. For example, the architecture included a trusted entity (like ERSE or IMT), the anonymous vending machine, and the price broadcaster. Three possible protocols were proposed for the payment of the energy transactions with different approaches. These protocols keep the privacy of the users, are safe and have an acceptable execution speed. They can also cope with inefficiency or energy transmission problems. In other words, these protocols have several properties that allow their use in real scenarios. The results obtained from tests show that the protocols are feasible to be put into practice.

As future work, other protocols could be drawn accordingly to the market model and laws, that are specific to each country. It should be noted that privacy is a feature that can be included without compromising the functionality of the system.

Bibliography

[1] Major power outage hits New York, other large cities. http://edition.cnn.com/2003/US/08/14/power.outage/, 2015. Accessed: 04-03-2015.

[2] Neon city goes dim as power shortage threatens traffic lights and telephones in Tokyo. http://www.news.com.au/world/neon-city-goes-dim-as-power-shortage-threatens-traffic-lights-and-telephones-in-tokyo/story-e6frfkyi-1226021645448, 2015. Accessed: 04-03-2015.

[3] ENTSO-E - European Network of Transmission System Operators for Electricity. https://www.entsoe.eu/db-query/consumption/mhlv-a-specific-country-every-3rd-wednesday-of-a-specific-year, 2015. Accessed: 04-03-2015.

[4] Ec type examination certificate. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/293328/SGS0043_Janz_C2801_Issue_5.doc.PDF, 2010. Accessed: 04-03-2015.

[5] Smart meter - Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Smart_meter, 2015. Accessed: 04-03-2015.

[6] Your Outlet Knows: How Smart Meters Can Reveal Behavior at Home, What We Watch on TV. http://www.bloomberg.com/news/2014-06-10/your-outlet-knows-how-smart-meters-can-reveal-behavior-at-home-what-we-watch-on-tv.html, 2015. Accessed: 04-03-2015.

[7] 'Smart' Meter Protests Spread as PG&E Officials Implicated in Spy Scandal. http://stopsmartmeters.org/2010/12/15/smart-meter-protests-spread-as-pge-officials-implicated-in-spy-scandal/, 2010. Accessed: 04-03-2015.

[8] Young Dae Ko and Young Jae Jang. The optimal system design of the online electric vehicle utilizing wireless power transmission technology. *IEEE Transactions on Intelligent Transportation Systems*, 14(3):1255–1265, 2013.

[9] Wirelessly charged electric vehicle runs in seoul. `http://www.thedetroitbureau.com/2010/03/wirelessly-charged-electric-vehicle-runs-in-seoul/`, 2010. Accessed: 04-05-2015.

[10] Wirelessly recharged electric bus rolls out. `http://www.korea.net/NewsFocus/Sci-Tech/view?articleId=111336`, 2013. Accessed: 04-05-2015.

[11] Introduction to Provable Security - Mahdi Zamani. `http://www.cs.unm.edu/~zamani/crypto/basics/`, 2015. Accessed: 04-03-2015.

[12] Alfredo Rial and George Danezis. Privacy-preserving smart metering. In *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society*, pages 49–60, 2011.

[13] Marek Jawurek, Martin Johns, and Florian Kerschbaum. Plug-in privacy for smart metering billing. In *Proceedings of the Privacy Enhancing Technologies*, pages 192–210. Springer, 2011.

[14] George Danezis, Markulf Kohlweiss, and Alfredo Rial. Differentially private billing with rebates. In *Information Hiding*, pages 148–162. Springer, 2011.

[15] Andres Molina-Markham, George Danezis, Kevin Fu, Prashant Shenoy, and David Irwin. Designing privacy-preserving smart meters with low-cost microcontrollers. In *Financial Cryptography and Data Security*, pages 239–253. Springer, 2012.

[16] Andrés Molina-Markham, Prashant Shenoy, Kevin Fu, Emmanuel Cecchet, and David Irwin. Private memoirs of a smart meter. In *Proceedings of the 2nd ACM workshop on Embedded Sensing Systems for Energy-efficiency in Building*, pages 61–66, 2010.

[17] Xiaojing Liao, David Formby, Carson Day, Raheem Beyah, et al. Towards secure metering data analysis via distributed differential privacy. In *44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, pages 780–785, 2014.

[18] Marek Jawurek and Florian Kerschbaum. Fault-tolerant privacy-preserving statistics. In *Proceedings of the Privacy Enhancing Technologies*, pages 221–238. Springer, 2012.

[19] Elaine Shi, T-H. Hubert Chan, Eleanor Rieffel, Richard Chow, and Dawn Song. Privacy preserving aggregation of time-series data. In *Network and Distributed System Security Symposium*, 2011.

[20] T-H Hubert Chan, Elaine Shi, and Dawn Song. Privacy-preserving stream aggregation with fault tolerance. In *Financial Cryptography and Data Security*, pages 200–214. Springer, 2012.

[21] Gergely Ács and Claude Castelluccia. I have a dream!(differentially private smart metering). In *Information Hiding*, pages 118–132. Springer, 2011.

[22] Jens-Matthias Bohli, Christoph Sorge, and Osman Ugus. A privacy model for smart metering. In *Proceedings of the IEEE International Conference on Communications Workshops*, pages 1–5, 2010.

[23] Costas Efthymiou and Georgios Kalogridis. Smart grid privacy via anonymization of smart metering data. In *Proceedings of the First IEEE International Conference on Smart Grid Communications*, pages 238–243, 2010.

[24] Tobias Jeske. Privacy-preserving smart metering without a trusted-third-party. In *Proceedings of the International Conference on Security and Cryptography*, pages 114–123. IEEE, 2011.

[25] Ronald Petrlic. A privacy-preserving concept for smart grids. *Sicherheit in vernetzten Systemen*, 18:B1–B14, 2010.

[26] Sushmita Ruj, Amiya Nayak, and Ivan Stojmenovic. A security architecture for data aggregation and access control in smart grids. *arXiv preprint arXiv:1111.2619*, 2011.

[27] Fengjun Li, Bo Luo, and Peng Liu. Secure information aggregation for smart grids using homomorphic encryption. In *Proceedings of the First IEEE International Conference on Smart Grid Communications*, pages 327–332, 2010.

[28] Klaus Kursawe, George Danezis, and Markulf Kohlweiss. Privacy-friendly aggregation for the smart-grid. In *Proceedings of the Privacy Enhancing Technologies*, pages 175–191. Springer, 2011.

[29] Zekeriya Erkin and Gene Tsudik. Private computation of spatial and temporal power consumption with smart meters. In *Applied Cryptography and Network Security*, pages 561–577. Springer, 2012.

[30] Flavio D Garcia and Bart Jacobs. Privacy-friendly energy-metering via homomorphic encryption. In *Security and Trust Management*, pages 226–238. Springer, 2011.

[31] Vibhor Rastogi and Suman Nath. Differentially private aggregation of distributed time-series with transformation and encryption. In *Proceedings of the ACM SIGMOD International Conference on Management of data*, pages 735–746, 2010.

[32] Cristina Rottondi, Giacomo Verticale, and Antonio Capone. A security framework for smart metering with multiple data consumers. In *Proceedings of the IEEE Conference on Computer Communications Workshops*, pages 103–108, 2012.

[33] Michael LeMay, George Gross, Carl Gunter, and Sanjam et al Garg. Unified architecture for large-scale attested metering. In *Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, pages 115–115. IEEE, 2007.

[34] Hsiao-Ying Lin, Wen-Guey Tzeng, Shiuan-Tzuo Shen, and Bao-Shuh P Lin. A practical smart metering system supporting privacy preserving billing and load monitoring. In *Applied Cryptography and Network Security*, pages 544–560. Springer, 2012.

[35] Georgios Kalogridis, Costas Efthymiou, Stojan Z. Denic, Tim Lewis, and Rafael Cepeda. Privacy for smart meters, towards undetectable appliance load signatures. In *Proceedings of the IEEE International Conference on Smart Grid Communications*, 2010.

[36] David Varodayan and Ashish Khisti. Smart meter privacy using a rechargeable battery: Minimizing the rate of information leakage. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1932–1935, 2011.

[37] Georgios Kalogridis, Costas Efthymiou, Stojan Z Denic, Tim Lewis, and Rafael et al Cepeda. Privacy for smart meters: Towards undetectable appliance load signatures. In *Proceedings of the First IEEE International Conference on Smart Grid Communications*, pages 232–237, 2010.

[38] Stephen McLaughlin, Patrick McDaniel, and William Aiello. Protecting consumer privacy from electric load monitoring. In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, pages 87–98, 2011.

[39] Gergely Acs, Claude Castelluccia, and William Lecat. Protecting against physical resource monitoring. In *Proceedings of the 10th annual ACM Workshop on Privacy in the Electronic Society*, pages 23–32, 2011.

[40] Ming-Chin Chuang and Jeng-Farn Lee. TEAM: Trust-extended authentication mechanism for vehicular ad hoc networks. *Systems Journal*, 8(3):749–758, 2014.

[41] Rasheed Hussain, Donghyun Kim, Michele Nogueira, Junggab Son, Alade O Tokuta, and Heekuck Oh. Pbf: A new privacy-aware billing framework for online electric vehicles with bidirectional auditability. *arXiv preprint arXiv:1504.05276*, 2015.

[42] Haojin Zhu, Xiaodong Lin, Rongxing Lu, Pin-han Ho, and Xuemin Sherman Shen. Slab: A secure localized authentication and billing scheme for wireless mesh networks. *IEEE Transactions on Wireless Communications*, 7(10):3858–3868, 2008.

[43] Hsin-Yi Lee and Yi-Bing Lin. Credit pre-reservation mechanism for UMTS prepaid service. *IEEE Transactions on Wireless Communications*, 9(6):1867–1873, 2010.

[44] Heekuck Oh. An Efficient Payment Method for Wireless Charging of Electrical Vehicles on Move. `http://infosec1.pusan.ac.kr/files/S1/S1-4.%204.An%20Efficient%20Payment%20Method%20for%20Wireless%20Charging%20of%20Electrical%20Vehicles%20on%20Move,Zeinab,%20Rasheed,%20Fizza,%20Ubaidullah,%20Heekuck%20Oh,%20%ED%95%9C%EC%96%91%EB%8C%80%ED%95%99%EA%B5%90.pdf`, 2015. Accessed: 04-05-2015.

[45] Lo-Yao Yeh and Yu-Cheng Lin. A proxy-based authentication and billing scheme with incentive-aware multihop forwarding for vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1607–1621, 2014.

[46] Lo-Yao Yeh and Jiun-Long Huang. Pbs: a portable billing scheme with fine-grained access control for service-oriented vehicular networks. *IEEE Transactions on Mobile Computing*, 13(11):2606–2619, 2014.

[47] Benessa Defend and Klaus Kursawe. Implementation of privacy-friendly aggregation for the smart grid. In *Proceedings of the ACM Workshop on Smart Energy Grid Security*, pages 65–74, 2013.

[48] Whitfield Diffie and Martin E Hellman. New directions in cryptography. *IEEE Transactions onInformation Theory*, 22(6):644–654, 1976.

[49] Ralph Charles Merkle. Secrecy, authentication, and public key systems. *Stanford University*, 1979.

[50] SHA-2 - Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/SHA-2`, 2015. Accessed: 04-03-2015.

[51] Daniel J Bernstein. Curve25519: new Diffie-Hellman speed records. In *Public Key Cryptography*, pages 207–228. Springer, 2006.

[52] Curve25519: high-speed elliptic-curve cryptography. http://cr.yp.to/ecdh.html, 2015. Accessed: 04-03-2015.

[53] curve25519-donna - A collection of implementations of curve25519, an elliptic curve Diffie Hellman primitive - Google Project Hosting. https://code.google.com/p/curve25519-donna/, 2015. Accessed: 04-03-2015.

[54] ckolivas/cgminer · GitHub. https://github.com/ckolivas/cgminer/, 2011. Accessed: 04-03-2015.

[55] Index of /apps/cgminer. http://ck.kolivas.org/apps/cgminer/, 2015. Accessed: 04-03-2015.

[56] David Chaum. Blind signatures for untraceable payments. In *Advances in cryptology*, pages 199–203. Springer, 1983.

[57] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash. In *Proceedings on Advances in cryptology*, pages 319–327. Springer-Verlag New York, Inc., 1990.

[58] Joan Daemen and Vincent Rijmen. *The design of Rijndael: AES-the advanced encryption standard*. Springer Science & Business Media, 2013.

[59] FIPS. 197, Advanced encryption standard (AES), national institute of standards and technology, us department of commerce (november 2001). *http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf*, 2001.

[60] GALS System Design: Side Channel Attack Secure Cryptographic Accelerators. http://www.iis.ee.ethz.ch/~kgf/acacia/c3.html, 2015. Accessed: 04-03-2015.

[61] William Stallings. *Network and internetwork security: principles and practice*, volume 1. Prentice Hall Englewood Cliffs, 1995.

[62] rsaAlgorithm.JPG. http://www.arl.wustl.edu/~jl1/education/cs502/images/rsa_algorithm.JPG, 2015. Accessed: 04-03-2015.

[63] Blind signature - Wikipedia. https://it.wikipedia.org/wiki/Blind_signature, 2015. Accessed: 04-03-2015.